



(51) International Patent Classification:

G09B 9/00 (2006.01) H04L 12/16 (2006.01)
G09B 9/08 (2006.01) G08G 5/00 (2006.01)

(21) International Application Number:

PCT/CA2017/050188

(22) International Filing Date:

15 February 2017 (15.02.2017)

(25) Filing Language:

English

(26) Publication Language:

English

(30) Priority Data:

2920913 17 February 2016 (17.02.2016) CA
15/046,103 17 February 2016 (17.02.2016) US

(71) Applicant: CAE INC. [CA/CA]; 8585 Cote-de-Liesse, Saint-Laurent, Québec H4T 1G6 (CA).

(72) Inventor: DELISLE, Jean-francois; 8585 Cote-de-Liesse, Saint-Laurent, Québec H4T 1G6 (CA).

(74) Agent: IP DELTA PLUS INC; 15-5585 Chemin Cote St-Luc, Montreal, Québec H3X 2C9 (CA).

(81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM,

AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:

— with international search report (Art. 21(3))

(54) Title: SIMULATION SERVER CAPABLE OF INTERACTING WITH A PLURALITY OF SERVERS

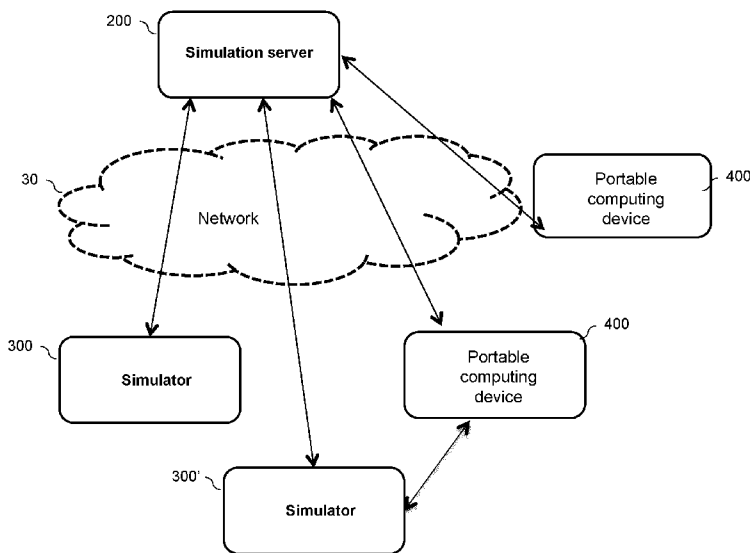


Figure 4

(57) Abstract: A simulation server capable of interacting with a plurality of simulators to perform a plurality of simulations. The simulation server comprises a communication interface for exchanging data with other entities. The processing server also comprises a processing unit for executing at least one simulation. The processing unit also generates simulator simulation data and transmits the simulator simulation data to at least one simulator via the communication interface. The simulator simulation data are representative of the execution of the at least one simulation. The processing unit also receives simulator interaction data from the at least one simulator via the communication interface. The processing unit further processes the simulator interaction data and controls the execution of the at least one simulation based on the processed simulator interaction data. The simulation server may also interact with one or more of portable computing devices to perform the plurality of simulations.

WO 2017/139880 A1

SIMULATION SERVER INTERACTING WITH A PLURALITY OF SERVERS

TECHNICAL FIELD

[0001] The present disclosure relates to the field of simulators. More specifically, the present disclosure relates a simulation server capable of interacting with a plurality of simulators to perform a plurality of simulations.

BACKGROUND

[0002] Flight simulators are used by commercial airlines and air forces to train their pilots to face various types of situations. A simulator is capable of simulating various functionalities of an aircraft, and of reproducing various operational conditions of a flight (e.g. takeoff, landing, hovering, etc.). A trainee (e.g. a pilot performing a training session) interacts with the simulator to control various functionalities of the simulated aircraft during a simulation executed by the simulator. Similarly, an instructor (e.g. an experienced pilot) may interact with the simulator for various purposes, including controlling a simulation currently executed by the simulator, creating or updating simulation scenarios, controlling the simulation environment of a trainee, etc.

[0003] A flight simulator is a very complex system, comprising dedicated hardware and software components for executing and controlling a simulation, as well as dedicated physical components (e.g. a simulated control cabin of an aircraft) for simulating corresponding physical components of the simulated aircraft. Thus, the design, usage, maintenance and upgrade of a flight simulator is very costly, in terms of financial resources as well as qualified human resources.

[0004] Since an increasing part of the added value of a flight simulator consists in its software components, the design of a flight simulator may benefit from the advances in client / server or cloud computing architectures. In particular,

some of the software components currently executed on a flight simulator may be executed on standardized servers or computing devices, to implement a client / server or cloud based simulation architecture. The advantages of a client / server or cloud based architecture in terms of costs, usage flexibility, ease of maintenance, etc. have been proven in various technological fields, and could be applied to the field of flight simulators, and more generally to other types of simulators (e.g. medical simulators).

[0005] Therefore, there is a need for a simulation server capable of interacting with a plurality of simulators to perform a plurality of simulations.

SUMMARY

[0006] According to a first aspect, the present disclosure provides a simulation server. The simulation server comprises a communication interface for exchanging data with other entities. The processing server also comprises a processing unit for executing at least one simulation. The processing unit also generates simulator simulation data and transmits the simulator simulation data to at least one simulator via the communication interface. The simulator simulation data are representative of the execution of the at least one simulation. The processing unit also receives simulator interaction data from the at least one simulator via the communication interface. The processing unit further processes the simulator interaction data and controls the execution of the at least one simulation based on the processed simulator interaction data.

[0007] In a particular aspect, the processing unit of the simulation server also generates and transmits portable computing device simulation data (representative of the execution of the at least one simulation) to at least one portable computing device, receives portable computing device interaction data from the at least one portable computing device, and processes the portable computing device interaction data to control the execution of the at least one

simulation based on the processed portable computing device interaction data.

BRIEF DESCRIPTION OF THE DRAWINGS

[0008] Embodiments of the disclosure will be described by way of example only with reference to the accompanying drawings, in which:

[0009] Figure 1 illustrates a legacy simulator configuration;

[0010] Figures 2A, 2B and 2C illustrate a simulation server interacting with a simulator and portable computing devices for executing a simulation;

[0011] Figures 3A, 3B and 3C respectively illustrate exemplary embodiments of components and functionalities of the simulation server, simulator and portable computing devices of Figures 2A, 2B and 2C;

[0012] Figure 4 illustrates a simulation server concurrently interacting with a plurality of simulators and portable computing devices;

[0013] Figure 5 illustrates respective displays of the simulator and portable computing device of Figures 3B and 3C;

[0014] Figures 6A, 6B and 6C represent an exemplary flow diagram illustrating interactions between components of the simulation server of Figure 3A and the portable computing device of Figure 3C;

[0015] Figures 7 and 8 illustrate components of a simulation server and portable computing device for managing lesson plans and associated events;

[0016] Figure 9 illustrates a KPI assigned to an event consisting of a pitch angle measurement; and

[0017] Figures 10 and 11 represent visual alarms or prediction indicators associated to an event being displayed on a visual representation of the execution of a simulation.

DETAILED DESCRIPTION

[0018] The foregoing and other features will become more apparent upon reading of the following non-restrictive description of illustrative embodiments thereof, given by way of example only with reference to the accompanying drawings. Like numerals represent like features on the various drawings.

[0019] Various aspects of the present disclosure generally address one or more of the problems related to the usage of a simulation server for providing simulations services to a plurality of simulators and portable computing devices. Although the examples provided in the rest of the disclosure are in the field of aircraft simulators, the teachings of the present disclosure can also be applied to simulators of terrestrial vehicles such as tanks, maritime vehicles such as boats, etc. The simulators may also perform a real time simulation of an underground system, a mining facility, a nuclear plant, a human body, etc.

SIMULATION SERVER

[0020] Referring now to Figure 1, a legacy simulator 100 is represented. The simulator 100 executes a simulation. The execution of the simulation is generally performed in real time and encompasses a plurality of functions, which are performed sequentially or concurrently.

[0021] The execution of the simulation comprises executing one or more simulation functionalities 110. In the case of an aircraft simulator, examples of simulation functionalities 110 include simulations of the following components of

the aircraft: simulation of the engines, simulation of the landing gear, simulation of the electrical circuits, simulation of the hydraulic circuits, simulation of the cockpit, etc. Furthermore, the execution of particular simulation functionality 110 may trigger the display of generated simulation data (e.g. in the form of a navigation map, a radar map, a weather map, a flight map, aircraft data, etc.) on a display of the simulator 100. Simulation functionalities 110 can be implemented separately in software modules executed by the simulator 100, or grouped per category of equipment, type of simulation, etc. into larger software modules. The simulator 100 is capable of executing several simulation functionalities 110 in parallel, to perform an exhaustive simulation of the aircraft. Alternatively, the simulator 100 executes a single simulation functionality 110 (or a limited number of simulation functionalities 110) to perform a restricted simulation of the aircraft, focusing on specific systems and sub-systems of the aircraft (e.g. only the engines, only the engines and landing gear in combination, etc.).

[0022] The execution of the simulation also comprises executing a simulator graphical user interface (GUI) functionality 130. The simulator GUI functionality 130 allows a user 20 (for example a trainee) to interact with the simulator 100, and more specifically with the simulation functionalities 110 currently executed by the simulator 100. In the case of an aircraft simulator, the simulator GUI functionality 130 comprises displaying simulation data generated by the simulation functionalities 110 on one or more displays of the simulator 100. The displayed simulation data may include flight parameters (e.g. altitude, speed, etc.), aircraft parameters (e.g. remaining fuel, alarms, etc.), maps (e.g. navigation map, weather map, radar map, etc.), virtual controls, out-of window information, etc. The simulator GUI functionality 130 also comprises receiving interactions from the user 20 via one or more user interfaces of the simulator 100. The user interface(s) may include traditional computer user interfaces (e.g. a keyboard, a mouse, a trackpad, a touch screen, etc.), as well as dedicated simulation user

interfaces (e.g. switches, simulation command controls, joysticks, etc.). The interactions received from the user 20 are processed by the simulation functionalities 110, and affect the simulation of one or more systems of the aircraft.

[0023] The execution of the simulation also comprises executing an Instructor Operating Station (IOS) functionality 120. The IOS functionality 120 allows a user 10 (for example an instructor) to interact with the simulator 100, and more specifically with the simulation functionalities 110 currently executed by the simulator 100. For instance, IOS User Interface (UI) pages are displayed on a display of the simulator 100, allowing the user 10 to control in real time the execution of a particular simulation scenario executed by the simulation functionalities 110. The IOS 120 comprises graphical control elements (e.g. menus and sub-menus, list boxes, etc.) for controlling the execution of the simulation (e.g. modifying simulation parameters) and graphical display elements (e.g. images, text fields, icons, embedded videos, etc.) for displaying simulation data generated by the simulation functionalities 110. In the case of an aircraft simulator, the user 10 interacts with the IOS 120 via one or more user interfaces (e.g. a keyboard, a mouse, a trackpad, a touch screen, etc.) to configure and / or update simulation parameters (e.g. weather conditions, flight plan, etc.). The configured / updated simulation parameters are processed by the simulation functionalities 110, and affect the simulation of one or more systems of the aircraft.

[0024] In a particular embodiment, the execution of the simulation also comprises executing at least one proxy function (not represented in Figure 1). The proxy function allows other functionalities of the simulator 100 (e.g. IOS functionality 120 and simulator GUI functionality 130) to interact with the simulation functionalities 110. A plurality of proxy functions may be executed concurrently, each proxy function providing an interface to a specific functionality

among the simulation functionalities 110.

[0025] Referring now concurrently to Figures 2A, 2B and 2C, embodiments of a simulation server 200 and a simulator 300 in accordance with the present invention are represented. The present simulation server 200 and simulator 300 rely on sharing processing performed when executing a simulation between themselves.

[0026] The simulation server 200 executes a simulation, which comprises executing one or more server simulation functionalities 210, executing a server IOS functionality 220, and executing a server graphical user interface (GUI) functionality 230. The simulator 300 also executes the simulation, which comprises executing one or more simulator simulation functionalities 310, executing a simulator IOS User Interface (UI) 320, and executing a simulator graphical user interface (GUI) 330.

[0027] The simulation functionalities 110 of the legacy simulator 100 have been split between the server simulation functionalities 210 and the simulator simulation functionalities 310. The server simulation functionalities 210 perform the computations for simulating the operations of the simulated entity. The present simulation server 200 and simulator 300 may be used to simulate any type of entity, such as for example aircrafts, spatial shuttles, vessels, vehicles, etc. For simplicity purposes, the following description will provide examples related to aircrafts. Such examples are provided for example purposes only, and should not be construed to limit the scope of protection sought for the present simulation server 200 and simulator 300.

[0028] The server simulation functionalities 210 may perform computations for simulating components of the vehicle being simulated, for example engines, electrical circuits, hydraulic circuits, information shown and displayed to a user of the simulator, etc. The server simulation functionalities 210 generate simulation data 211, which are used locally by other functionalities

implemented by the simulation server 200 (e.g. the server IOS functionality 220 and the server GUI functionality 230). The server simulation functionalities 210 also generate simulation data 212, which are transmitted to the simulator 300 through a communication network 30. The transmitted simulation data 212 can be used by the simulator GUI 330, the simulator IOS UI 320 and the simulator simulation functionalities 310, but may also be used by other functionalities implemented by the simulator 300 (this use case is not represented in Figure 2A). The simulation data 212 include visual data in the form of static 2D or 3D simulation images, or a flow of 2D or 3D simulation images. Examples of visual data include a navigation map, a radar map, a weather map, a flight map, aircraft data, etc. For instance, a map may consist in a static 2D or 3D image, or a flow of 2D or 3D images corresponding to a real time streaming of the map. The visual data are displayed by at least one simulator GUI 330 and simulator IOS UI 320 on display(s) of the simulator 300. The simulation data 212 may also include actuation data, which are processed by the simulator simulation functionalities 310 for actuating physical component(s) of the simulator 300 (e.g. actuating hydraulic cylinders for moving a simulated control cabin representative of the entity). The simulator simulation functionalities 310 may also perform local computations for simulating some of the operations of the simulated entity, which are not performed by the server simulation functionalities 210. The simulator simulation functionalities 310 also generate feedback simulation data 212, which are transmitted to the simulation server 200 through the communication network 30. These feedback simulation data 212 are representative of an operational status of the simulator 300, and are used by the server simulation functionalities 210 as a feedback for taking into consideration the operational status of the simulator 300 when executing the simulation. For example, the server simulation functionalities 210 transmit simulation data 212 for actuating (by the simulator simulation functionalities 310) a physical component of the simulator 300, and the simulator simulation functionalities 310 transmit feedback simulation data 212

indicating that the activation of the physical component of the simulator 300 has been completed. Upon reception of the feedback simulation data 212, the server simulation functionalities 210 proceed with the next step in the execution of the simulation, taking into consideration the new operational status of the simulator 300.

[0029] The IOS functionality 120 of the legacy simulator 100 has also been split between the server IOS functionality 220 and the simulator IOS UI 320. The server IOS functionality 220 performs the computations related to the IOS functionality, while the simulator IOS UI 320 performs the interactions with a user 10. The server IOS functionality 220 generates IOS control and monitoring data 222, which are transmitted to the simulator 300 through the communication network 30. The IOS control and monitoring data 222 are used by the simulator IOS UI 320 for interacting with the user 10. The IOS control and monitoring data 222 includes IOS UI page(s), e.g. each IOS UI page corresponding to a user interface with sub-groups of instances, which are displayed on a display of the simulator 300 by the simulator IOS UI 320. The displayed IOS UI page(s) allow the user 10 to control in real time (at the simulator 300) the execution of a particular simulation scenario executed by the server simulation functionalities 210. More specifically, by interacting with the IOS UI 320, the user 10 generates IOS interaction data 222 (e.g. configuration, monitoring and / or update of simulation parameters), which are transmitted via the network 30 to the server IOS functionality 220. The server IOS functionality 220 interacts with the server simulation functionalities 210 by exchanging data 211. As mentioned previously, the data 211 include simulation data generated by the server simulation functionalities 210, and used by the server IOS functionality 220 for generating IOS UI pages(s), which are transmitted to the simulator IOS UI 320. The data 211 also include the configuration / update of simulation parameters received from the simulator IOS UI 320, which may be pre-processed by the server IOS functionality

220 before transmission to the server simulation functionalities 210. The server simulation functionalities 210 use the configuration / update of simulation parameters to control the execution of the simulation. As mentioned previously, the simulator IOS UI 320 also receives simulation data 212 (e.g. 2D or 3D maps) from the server simulation functionalities 210, displayed on display(s) of the simulator 300.

[0030] The simulator GUI functionality 130 of the legacy simulator 100 has also been split between the server GUI functionality 230 and the simulator GUI 330. The server GUI functionality 230 performs the computations related to the graphical representations to be displayed to the user 20, while the simulator GUI 330 performs the interactions with the user 20. The server GUI functionality 230 generates graphical user interface (GUI) graphical data 232, which are transmitted to the simulator 300 through the communication network 30. The GUI graphical data 232 are used by the simulator GUI 330 for interacting with the user 20. The GUI graphical data 232 include simulation parameters, events and information (e.g. entity parameters, events, maps, etc.), which are displayed on display(s) of the simulator 300 by the simulator GUI 330. The displayed GUI graphical data 232 allow the user 20 to interact in real time (at the simulator 300) with the execution of a particular simulation scenario executed by the server simulation functionalities 210. More specifically, based on the displayed GUI graphical data 232, the user 20 interacts with one or more user interfaces of the simulator 300. The simulator GUI 330 generates GUI interaction data 232 corresponding to the user 20 interactions, which are transmitted via the network 30 to the server GUI functionality 230. The server GUI functionality 230 interacts with the server simulation functionalities 210 by exchanging data 211. As mentioned previously, the data 211 include simulation data generated by the server simulation functionalities 210, and used by the server GUI functionality 230 for generating the simulation parameters, events and information transmitted to

the simulator GUI 330. The data 211 also include the user 20 interactions received from the simulator GUI 330, which may be pre-processed by the server GUI functionality 230 before transmission to the server simulation functionalities 210. The server simulation functionalities 210 use the user 20 interactions to control the execution of the simulation. As mentioned previously, the simulator GUI 330 also receives simulation data 212 (e.g. 2D or 3D maps) from the server simulation functionalities 210, which are further displayed on the simulator 300 through the simulator GUI 330.

[0031] In a particular embodiment, at least one of the functionalities implemented by the legacy simulator 100 of Figure 1 may not be deported on the simulation server 200, but fully implemented on the simulator 300. For example, the simulation server 200 does not include the server GUI functionality 230 and the simulator 300 includes the simulator GUI functionality 130 of the legacy simulator 100 in place of the simulator GUI 330. In this particular embodiment, the simulator GUI 130 is implemented on the simulator 300 and interacts directly with the server simulation functionalities 210 via the network 30. This embodiment is not represented in the Figures.

[0032] The network 30 may consist of a mobile network (e.g. a Wi-Fi network or cellular network), a fixed network (e.g. an Ethernet network), a combination thereof, etc. The network 30 may allow communications between devices over an Intranet, an Extranet, the global Internet, etc. The simulation server 200 and the simulator 300 both include a communication interface compatible with the network 30, for exchanging data over the network 30. For example, the simulation server 200 and the simulator 300 comprise a communication interface supporting both Wi-Fi and Ethernet, to easily adapt to a particular network 30 deployed at the premises where they are respectively operating.

[0033] Although not represented in Figure 2A for simplification purposes,

the server simulation functionalities 210 generally comprise a plurality of functions, for example a Weather function and a Navigation function (represented and described later with respect to Figure 3A). The Navigation function of the server simulation functionalities 210 interfaces with other functionalities of the simulation server 200 (e.g. with the server GUI functionality 230 and with a dedicated IOS function IOS_1 (represented in Figure 3A) of the server IOS functionality 220). Similarly, the Weather function of the server simulation functionalities 210 interfaces with other functionalities of the simulation server 200 (e.g. with the server GUI functionality 230 and with a dedicated IOS function IOS_2 (represented in Figure 3A) of the server IOS functionality 220). As mentioned previously with respect to the legacy simulator represented in Figure 1, a plurality of proxy functions may be used to interface the functions (e.g. Navigation and Weather respectively) of the server simulation functionalities 210 with other functionalities of the simulation server 200 (e.g. the server IOS functionality 220 and the server GUI functionality 230).

[0034] In the rest of the description, the simulation data 212 generated and transmitted by the server simulation functionalities 210 to the simulator 300 may be referred to as simulator simulation data. The IOS control and monitoring data 222 generated and transmitted by the server IOS functionality 220 to the simulator 300 may be referred to as simulator IOS control and simulation data. The GUI graphical data 232 generated and transmitted by the server GUI functionality 230 to the simulator 300 may be referred to as simulator graphical data.

[0035] The IOS interaction data 222 generated and transmitted by the simulator IOS UI 320 to the simulation server 200 and the GUI interaction data 232 generated and transmitted by the simulator GUI 330 to the simulation server 200 may be referred to as simulator interaction data.

[0036] Referring now concurrently to Figures 2A and 2B, portable

computing devices 400 and 400' are represented in Figure 2B. The portable computing devices 400 and 400' implement some of the functionalities implemented on the simulator 300 represented in Figure 2A.

[0037] The portable computing device 400 and 400' may consist in various types of computing devices having a form factor allowing easy carrying. Examples of such portable computing devices 400 and 400' include laptops, tablets, etc. The portable computing devices 400 and 400' both include a communication interface compatible with the network 30, for exchanging data with the simulation server 200 over the network 30.

[0038] The portable computing device 400 implements a portable computing device IOS UI 420 similar to the simulator IOS UI 320. The portable computing device IOS UI 420 exchanges IOS data 222 (IOS control and monitoring data, and IOS interaction data) with the server IOS functionality 220 over the network 30, in a similar manner as the simulator IOS UI 320. The portable computing device IOS UI 420 allows the user 10 to control in real time, via the portable computing device 400, the execution of a particular simulation scenario executed by the server simulation functionalities 210. The portable computing device IOS UI 420 also receives simulation data 212 (e.g. 2D or 3D maps) from the server simulation functionalities 210, which are further displayed on the portable computing device 400 through the portable computing device IOS UI 420.

[0039] In this configuration, the IOS functionality 120 of the legacy simulator 100 represented in Figure 1 has been split between the server IOS functionality 220 and the portable computer device IOS UI 420. The server IOS functionality 220 performs the computations related to the IOS functionality, while the portable computing device IOS UI 420 performs the interactions with the user 10.

[0040] The server IOS functionality 220 is adapted for simultaneously or

alternatively supporting interactions with the simulator IOS UI 320 and the portable computing device IOS UI 420. For example, the same user 10 can use the simulator IOS UI 320 during a first simulation session, and the portable computing device IOS UI 420 during a second simulation session, both sessions being performed by executing server simulation functionalities 210. In another example, a first user 10 uses the simulator IOS UI 320 during a simulation session (for controlling a first simulation functionality), and a second user 10 simultaneously uses the portable computing device IOS UI 420 during the same simulation session (for controlling a second simulation functionality), the simulation session being performed by executing the server simulation functionalities 210.

[0041] The portable computing device 400' implements a portable computing device graphical user interface (GUI) 430 similar to the simulator GUI 330. The portable computing device GUI 430 exchanges GUI graphical data and GUI interaction data 232 with the server GUI functionality 230 over the network 30, in a similar manner as the simulator GUI 330. The portable computing device GUI 430 allows the user 20 to interact in real time, via the portable computing device 400', with the execution of a particular simulation scenario executed by the server simulation functionalities 210. As mentioned previously, the portable computing device GUI 430 also receives and displays simulation data 212 (e.g. 2D or 3D maps) from the server simulation functionalities 210.

[0042] In this configuration, the simulator GUI functionality 130 of the legacy simulator 100 represented in Figure 1 has been split between the server GUI functionality 230 and the portable computing device GUI 430. The server GUI functionality 230 performs the computations related to the graphical representations to be displayed to the user 10, while the portable computing device GUI 430 performs the interactions with the user 20.

[0043] The server GUI functionality 230 is adapted for simultaneously or

alternatively supporting interactions with the simulator GUI 330 and the portable computing device GUI 430. For example, user 20 can use the simulator GUI 330 during a first training session, and the portable computing device GUI 430 during a second training session, both simulation sessions being performed by executing server simulation functionalities 210. In another example, a first user 20 uses the simulator GUI 330 during a simulation session (for interacting with a first simulation functionality 210), and a second user 20 simultaneously uses the portable computing device GUI 430 during the same simulation session (for interacting with a second simulation functionality 210), the simulation sessions being performed by executing the first and second server simulation functionalities 210.

[0044] In the rest of the description, the simulation data 212 generated and transmitted by the server simulation functionalities 210 to a portable computing device 400 or 400' may be referred to as portable computing device simulation data.

[0045] The IOS control and monitoring data 222 generated and transmitted by the server IOS functionality 220 to a portable computing device 400 may be referred to as portable computing device IOS control and monitoring data. The GUI graphical data 232 generated and transmitted by the server GUI functionality 230 to a portable computing device 400' may be referred to as portable computing device graphical data. The IOS interaction data 222 generated and transmitted by the portable computing device IOS UI 420 to the simulation server 200 and the GUI interaction data 232 generated and transmitted by the portable computing device GUI 430 to the simulation server 200 may be referred to as portable computing device interaction data.

[0046] Figure 2C illustrates another operational configuration of the simulation server 200 and simulator 300. In this particular configuration, the user 10 uses the portable computing device IOS UI 420 for controlling in real time

(through the server IOS functionality 220) the execution of a particular simulation scenario executed by the server simulation functionalities 210. The user 20 uses the simulator GUI 330 for interacting in real time (through the server GUI functionality 230) with the execution of the same particular simulation scenario executed by the server simulation functionalities 210.

[0047] A person skilled in the art will readily understand that the usage of the simulation server 200 provides the required flexibility for implementing a plurality of other operational configurations involving the simulation server 200, the simulator 300 and portable computing device(s) (e.g. 400 and 400'), which have not been represented in the Figures for simplification purposes.

[0048] Referring now to Figures 3A, 3B and 3C, exemplary embodiments of components and functionalities of the simulation server 200, the simulator 300 and the portable computing device 400 are represented.

[0049] Referring more specifically to Figure 3A, the simulation server 200 comprises a processing unit 201, having one or more processors (not represented in Figure 3A for simplification purposes) capable of executing instructions of computer program(s). Each processor may further have one or several cores. The processing unit 201 implements functionalities of the simulation server 200 by executing computer program(s) instructions. The functionalities implemented by the processing unit 201 include the server simulation functionalities 210, the server IOS functionality 220 and the server GUI functionality 230.

[0050] The simulation server 200 comprises memory 202 for storing instructions of the computer program(s) executed by the processing unit 201, data generated by the execution of the computer program(s), data received via a communication interface 203, etc. The simulation server 200 may comprise several types of memories, including volatile memory, non-volatile memory, etc.

[0051] The simulation server 200 comprises the communication interface 203, for exchanging data with other devices, including one or more simulators 300 and / or one or more portable computing devices 400 via the network 30. The exchanged data comprise the IOS data 222, the GUI data 232 and the simulation data 212 represented in Figures 2A and 2B. The communication interface 203 supports one of more communication protocols, such as Wi-Fi, Ethernet, etc.

[0052] The simulation server 200 may comprise a display 204 (e.g. a regular screen or a tactile screen) for displaying data processed and / or generated by the processing unit 201. The simulation server 200 may also comprise one or more user interface 205 (e.g. a mouse, a keyboard, a trackpad, a touchscreen, etc.) for allowing a user to interact directly with the simulation server 200.

[0053] The server IOS functionality 220, the server simulation functionalities 210, and the server GUI functionality 230 are implemented by one or more computer programs. Each computer program comprises instructions for implementing the corresponding function when executed by the processing unit 201. The instructions are comprised in a non-transitory computer program product (e.g. memory 202). The instructions are deliverable via an electronically-readable media, such as a storage media (e.g. a USB key or a CD-ROM) or the network 30 (through the communication interface 203).

[0054] Referring more specifically to Figure 3B, the simulator 300 comprises a processing unit 301, having one or more processors (not represented in Figure 3B for simplification purposes) capable of executing instructions of computer program(s). Each processor may further have one or several cores. The processing unit 301 implements functionalities of the simulator 300 by executing instructions of the computer program(s). The functionalities implemented by the processing unit 301 include the simulator simulation functionalities 310, the simulator IOS UI 320 and the simulator GUI 330.

[0055] The simulator 300 comprises memory 302 for storing instructions of the computer program(s) executed by the processing unit 301, data generated by the execution of the computer program(s), data received via a communication interface 303, etc. The simulator 300 may comprise several types of memories, including volatile memory, non-volatile memory, etc.

[0056] The simulator 300 comprises the communication interface 303, for exchanging data with other devices, including the simulation server 200 via the network 30. The exchanged data comprise the IOS data 222, the GUI data 232 and the simulation data 212 represented in Figure 2A. The communication interface 303 supports one or more communication protocols, such as Wi-Fi, Ethernet, etc.

[0057] The simulator comprises one or more actuators 306 for actuating physical component(s) of the simulator 300 (for example. actuating hydraulic cylinders for moving a simulated control cabin of an aircraft) under the control of the simulator simulation functionalities 310 executed by the processing unit 301.

[0058] The simulator 300 comprises one or more displays 304 (e.g. a regular screen or a tactile screen) for displaying data processed and / or generated by the processing unit 301. The simulator 300 also comprises one or more user interface 305 (e.g. traditional computer user interfaces as well as dedicated simulation user interfaces) for allowing users to interact with the simulator 300.

[0059] In a particular embodiment, the simulator IOS UI 320 includes a display function 321 and an interaction function 322. The display function 321 processes the IOS control and monitoring data 222 and the simulation data 212 of Figure 2A, received from the simulation server 200 via the communication interface 303, and displays the processed IOS control and monitoring data 222 and simulation data 212 on the display 304.

[0060] The simulator IOS UI interaction function 322 generates the IOS interaction data 222 of Figure 2A based on the interactions of the user 10 (via the user interface(s) 305) with the processed IOS control and monitoring data 222 displayed on the display 304. The IOS interaction data 222 are transmitted to the server IOS functionality 220 of the simulation server 200 through the communication interface 303.

[0061] The simulator GUI 330 may also include a display function and an interaction function, not represented in Figure 3B for simplification purposes. The display function processes the GUI graphical data 232 and the server simulation data 212 of Figure 2A, received from the simulation server 200 via the communication interface 303, and displays the processed GUI graphical data 232 and server simulation data 212 on the display 304.

[0062] The interaction function of the simulator GUI 330 generates the GUI interaction data 232 of Figure 2A based on the interactions of the user 20 (via the user interface(s) 305) with the processed GUI graphical data 232 displayed on the display 304. The GUI interaction data 232 are transmitted to the server GUI functionality 230 of the simulation server 200 through the communication interface 303.

[0063] In another particular embodiment, the simulator simulation functionalities 310 include a display function 311 and an actuation function 312. The display function 311 processes simulation data 212 of Figure 2A received from the simulation server 200 via the communication interface 303, and displays the processed simulation data on the display 304. The display function 311 can be used for displaying the simulation data 212 of Figure 2A which cannot be displayed via the simulator IOS UI 320 or the simulator GUI 330.

[0064] The actuation function 312 processes other simulation data 212 of Figure 2A which include actuation data, received from the simulation server 200 via the communication interface 303. The actuation function 312 processes the

received actuation data and actuates a corresponding actuator 306. The actuation function 312 also generates feedback actuation data representative of a state of the actuated corresponding actuator 306. The feedback actuation data are transmitted to the simulation server 200 through the communication interface 303. More specifically, simulation data 212 of Figure 2A comprising the feedback actuation data are transmitted to the server simulation functionalities 210 represented in Figure 3A.

[0065] The simulator simulation functionalities 310, the simulator IOS UI 320, and the simulator GUI 320 are implemented by one or more computer programs. Each computer program comprises instruction for implementing the corresponding function when executed by the processing unit 301. The instructions are comprised in a non-transitory computer program product (e.g. memory 302). The instructions are deliverable via an electronically-readable media, such as a storage media (e.g. a USB key or a CD-ROM) or the network 30 (through the communication interface 303).

[0066] Referring more specifically to Figure 3C, the portable computing device 400 comprises a processing unit 401, having one or more processors (not represented in Figure 3C for simplification purposes) capable of executing instructions of computer program(s). Each processor may further have one or several cores. The processing unit 401 implements functionalities of the portable computing device 400 by executing instructions of the computer program(s). The functionalities implemented by the processing unit 401 include the portable computing device IOS UI 420.

[0067] The portable computing device 400 comprises memory 402 for storing instructions of the computer program(s) executed by the processing unit 401, data generated by the execution of the computer program(s), data received via a communication interface 403, etc. The portable computing device 400 may comprise several types of memories, including volatile memory, non-volatile

memory, etc.

[0068] The portable computing device 400 comprises the communication interface 403, for exchanging data with other devices, including the simulation server 200 via the network 30. The exchanged data comprise the IOS data 222 and the simulation data 212 represented in Figure 2B. The communication interface 403 supports one of more communication protocols, such as Wi-Fi, Ethernet, etc.

[0069] The portable computing device 400 comprises a display 404 (e.g. a regular screen or a tactile screen) for displaying data processed and / or generated by the processing unit 401. The portable computing device 400 also comprises at least one user interface 405 (e.g. a mouse, a keyboard, a trackpad, a touchscreen, etc.) for allowing a user (not represented in Figure 3C) to interact with the portable computing device 400.

[0070] In a particular embodiment, the portable computing device IOS UI 420 includes a display function 421 and an interaction function 422. The display function 421 and interaction function 422 operate in a similar manner as the display function 321 and interaction function 322 represented in Figure 3B.

[0071] The portable computing device IOS UI 420 is implemented by one or more computer programs. Each computer program comprises instruction for implementing the corresponding function when executed by the processing unit 401. The instructions are comprised in a non-transitory computer program product (e.g. memory 402). The instructions are deliverable via an electronically-readable media, such as a storage media (e.g. a USB key or a CD-ROM) or the network 30 (through the communication interface 403).

[0072] In an alternative embodiment not represented in the Figures for simplification purposes, the portable computing device 400 of Figure 3C may correspond to the portable computing device 400' represented in Figure 2B. The

functionalities implemented by the processing unit 401 include the portable computing device GUI 430 of Figure 2B in place of the portable computing device IOS UI 420. The data exchanged with the simulation server 200 comprise the GUI data 232 and the simulation data 212 of Figure 2B.

[0073] In still another alternative embodiment not represented in the Figures for simplification purposes, the portable computing device 400 of Figure 3C may be adapted for implementing both the portable computing device IOS UI 420 and the portable computing device GUI 430 illustrated in Figure 2B. If user 10 is using the portable computing device 400, the portable computing device IOS UI 420 is executed by the processing unit 401. If user 20 is using the portable computing device 400, the portable computing device GUI 430 illustrated in Figure 2B is executed by the processing unit 401.

[0074] In a particular aspect, the processing unit 201 of the simulation server 200 further executes a web server function 250, for implementing the exchange of data between the simulation server 200 and the simulator 300 or the portable computing devices 400. One of the users 10 or 20 (for example an instructor or a trainee) initiates a simulation web session between the web server function 250 and a web client implemented by the simulator 300 or portable computing devices 400, as will be detailed later in the description. Once the simulation web session is set up, data can be exchanged between the simulation server 200 and the simulator 300 or portable computing devices 400 via the web server function 250. The simulation data 212, the IOS data 222, and the GUI data 232 represented in Figures 2A and 2B are exchanged via the web server function 250.

[0075] In another particular aspect, the processing unit 201 of the simulation server 200 further executes at least one rendering function 240. Each instance of rendering function 240 is launched by the web server function 250, after the aforementioned simulation web session has been initiated, as will be

detailed later in the description. Each instance of rendering function 240 generates data adapted for rendering on a specific destination device. For example, if the specific destination device is the portable computing device 400, data such as 2D or 3D simulation images may need to be adapted to the particular processing and display capabilities of the portable computing device 400. If the specific destination device is the simulator 300, these data may not need a specific adaptation for rendering on the simulator 300. Thus, data which do not need to be adapted are transmitted directly by the web server function 250, while data which need to be adapted (e.g. 2D or 3D simulation images) are processed by an instance of rendering function 240 for adaptation purposes. The adapted data may be transmitted by the rendering function 240 to a destination device. Alternatively, the adapted data are also transmitted via the web server function 250 to a destination device. For illustration purposes, in the rest of the description, the adapted data will be transmitted by the rendering function 240 to a destination device (e.g. portable computing device 400). Usually, some of the simulation data 212 represented in Figures 2A and 2B contain 2D or 3D simulation images such as maps, which need to be adapted by an instance of rendering function 240, in particular for transmission to a portable computing device 400 with limited processing capabilities, display capabilities, etc. However, some of the IOS data 222 and GUI data 232 represented in Figures 2A and 2B may also be adapted by an instance of rendering function 240 when needed.

[0076] The adaptation of a static 2D or 3D simulation image (or a flow of 2D or 3D simulation images) by an instance of rendering function 240 may consist in one of the following: encoding the simulation image(s) in a particular format (e.g. JPEG, GIF, TIFF, PNG, etc.), applying a particular compression algorithm to the simulation image(s), applying a particular image sampling algorithm to the simulation image(s), and applying a particular algorithm for lowering an image resolution of the simulation image(s).

[0077] Since at least some of the communications with the simulation server 200 are performed through the web server function 250, the processing unit 301 of the simulator 300 and the processing unit 401 of the portable computing device 400 respectively execute a web client function 350 and 450. The web client functions 350 and 450 are used for establishing the previously described simulation web session with the web server function 250 of the simulation server 200. The web client functions 350 and 450 are further used for exchanging data (e.g. simulation data 212, IOS data 222 and GUI data 232 represented in Figures 2A and 2B) with the web server function 250 of the simulation server 200. For instance, data transmitted by the web server function 250 to the simulator IOS UI 320 are received by the web client function 350, and forwarded to the simulator IOS UI 320. Data to be transmitted by the simulator IOS UI 320 to the simulation server 200 are transmitted to the web client function 350, which forwards them to the web server function 250. The use of a web client for exchanging data is well known in the art, and will not be further detailed in the rest of the description.

[0078] With respect to the adapted data transmitted by an instance of rendering function 240 of the simulation server 200, they are not received by a web client (e.g. 450 on the portable computing device 400), but received directly by the destination functionality (e.g. portable computing device IOS UI 420). A proprietary communication protocol using pre-defined or dynamically allocated communication sockets can be used for transmitting the adapted data, as is well known in the art.

[0079] For illustration purposes, the operations of the web server function 250 and the rendering functions 240 will now be detailed in the context of an exchange of data between the simulation server 200 and the portable computing device 400. In this context, some of the data transmitted by the simulation server 200 need to be adapted to the capabilities of the portable

computing device 400 by one or more instances of rendering function 240, while other data can be transmitted directly via the web server function 250.

[0080] The data adapted for rendering on a particular destination device (e.g. portable computing device 400) usually consist in simulation data generated by the simulation functionalities 210 (e.g. 2D or 3D maps), and adapted by a particular instance of rendering function 240. For example, the aforementioned Navigation function of the server simulation functionalities 210 generates simulation data transmitted to a Navigation rendering function 240. The Navigation rendering function 240 adapts the simulation data (e.g. a navigation map) for rendering on the portable computing device 400, and transmits the adapted simulation data to the portable computing device 400. Similarly, the aforementioned Weather function of the server simulation functionalities 210 generates simulation data transmitted to a Weather rendering function 240. The Weather rendering function 240 adapts the simulation data (e.g. a weather map) for rendering on the portable computing device 400, and transmits the adapted simulation data to the portable computing device 400. A plurality of instances of rendering function 240 (e.g. Navigation and Weather rendering functions) can be simultaneously generating and transmitting adapted simulation data to the portable computing device 400. The display function 421 of the portable computing device IOS UI 420 receives and displays the simulation data which have been adapted and transmitted by the Navigation and Weather rendering functions 240.

[0081] The web server function 250 directly transmits data to the portable computing device 400, which do not need to be processed by one of the instances of rendering function 240. Such data usually include the IOS control and monitoring data 222 (e.g. a control web page) generated by the server IOS functionality 220, which are transmitted to the web server function 250, and further transmitted to the portable computing device 400. Such data may also include

complementary simulation data generated by one of the server simulation functionalities 210. For instance, the Navigation function of the server simulation functionalities 210 generates complementary simulation data (e.g. parameters of the simulation such as wind speed, events of the simulation such as aircraft speed too high, etc.), which are transmitted to the web server function 250, and further transmitted to the portable computing device 400. The parameters and / or events can be displayed on the display 404 of the portable computing device 400 in the form of icons, text fields, etc. For instance, the parameters and / or events constitute additional simulation information displayed in complement of a Navigation map displayed on the display 404 of the portable computing device 400. The Navigation map has been generated by the Navigation rendering function 240 based on simulation data generated by the Navigation function of the server simulation functionalities 210, and transmitted to the portable computing device 400.

[0082] For example, the IOS function IOS_1 of the server IOS functionality 220 transmits IOS control and monitoring data 222 (e.g. a Navigation control web page) to the portable computing device 400 for controlling the execution of the Navigation function of the server simulation functionalities 210. The IOS control and monitoring data 222 are transmitted by the IOS function IOS_1 to the web server function 250, and further transmitted to the portable computing device 400 for display. The IOS control and monitoring data 222 are displayed by the display function 421 of the portable computing device IOS UI 420 on the display 404. User 10 interacts with the displayed IOS control and monitoring data 222 (e.g. the Navigation control web page) and the interaction function 422 of the portable computing device IOS UI 420 generates IOS interaction data 222. User 10 interacts via the user interface 405 (e.g. a keyboard, a mouse, a trackpad, a touch screen, etc.) of the portable computing device 400, and the IOS interaction data 222 are generated based on this interaction. The IOS

interaction data 222 are transmitted by the portable computing device 400 to the web server function 250, and forwarded to the IOS function IOS_1. The IOS function IOS_1 processes the IOS interaction data 222 and controls the execution of the Navigation function of the server simulation functionalities 210 based on the processed IOS interaction data 222.

[0083] Simultaneously, the IOS function IOS_2 of the server IOS functionality 220 transmits IOS control and monitoring data 222 (e.g. a Weather control web page) to the portable computing device 400 for controlling the execution of the Weather function of the server simulation functionalities 210. The IOS control and monitoring data 222 are transmitted by the IOS function IOS_2 to the web server function 250, and further transmitted to the portable computing device 400 for display. The IOS control and monitoring data 222 are displayed by the display function 421 of the portable computer device IOS UI 420 on the display 404. User 10 interacts with the displayed control data (e.g. the Weather control web page) and the interaction function 422 of the portable computing device IOS UI 420 generates IOS interaction data 222. The IOS interaction data 222 are transmitted by the portable computing device 400 to the web server function 250, and forwarded to the IOS function IOS_2. The IOS function IOS_2 processes the IOS interaction data 222 and controls the execution of the Weather function of the server simulation functionalities 210 based on the processed IOS interaction data 222.

[0084] In a particular embodiment, a visual database (not shown in the Figures) may be used on the simulation server 200. The visual database contains data (e.g. terrain, buildings, 3D models, etc.) that can be streamed and displayed on the portable computing device 400, via one or more instances of rendering function 240. The visual database also contains parameters and / or events that can be overlaid on the displayed data, after direct transmission to the portable computing device 400 via the web server function 250.

[0085] Referring now simultaneously to Figures 3A and 4, Figure 4 illustrates a single simulation server 200 supporting a plurality of simulators (e.g. 300 and 300'), and a plurality of computing devices (e.g. 400 and 400'). Although two simulators have been represented in Figure 4, the simulation server 200 may support any number of simulators, based on its processing and communication capabilities. Similarly, although two portable computing devices have been represented in Figure 4, the simulation server 200 may support any number of portable computing devices, based on its processing and communication capabilities.

[0086] For illustration purposes, the simulation server 200 executes a first simulation and a first user (a trainee in this particular instance) interacts with the first simulation via the simulator 300, while a second user (an instructor in this particular instance) interacts with the first simulation via the portable computing device 400. Simultaneously, the simulation server 200 executes a second simulation and a third user (another trainee) interacts with the second simulation via the simulator 300', while a fourth user (a second instructor) interacts with the second simulation via the portable computing device 400'.

[0087] For illustration purposes, the server simulation functionalities 210 of the simulation server 200 execute simultaneously a first instance of Weather function in relation to the first simulation, and a second instance of Weather function in relation to the second simulation. Each instance of Weather function generates simulation data (e.g. a weather map), which are respectively transmitted directly to the simulators 300 and 300', via the web server function 250, without using a rendering function 240. Although two instances of Weather function are mentioned in this example, a larger number of instances can be operating simultaneously on the simulation server 200.

[0088] For illustration purposes, the rendering functions 240 execute simultaneously a first instance of Weather rendering function in relation to the first

simulation, and a second instance of Weather rendering function in relation to the second simulation. Each instance of Weather function executed by the server simulation functionalities 210 generates simulation data (e.g. the weather map), which are respectively adapted by the instances of Weather rendering function, before transmission to the portable computing devices 400 and 400'. Although two instances of Weather rendering function are mentioned in this example, a larger number of instances can be operating simultaneously on the simulation server 200. The adapted simulation data transmitted to the portable computing devices 400 and 400' may differ, based on specific characteristics of each of the portable computing devices 400 and 400'. For instance, a Weather map with a better resolution may be generated for the portable computing devices 400 than for the portable computing device 400'.

[0089] For illustration purposes, the server IOS functionality 220 executes simultaneously a first and a second instance of the IOS function IOS_2 for respectively controlling the first and second instances of Weather function executed by the server simulation functionalities 210. The first instance of IOS function IOS_2 transmits IOS control and monitoring data 222 (e.g. a Weather control web page) to the portable computing device 400 for controlling the execution of the first instance of Weather function. The control data are transmitted by the first instance of IOS function IOS_2 to the web server function 250, and further transmitted to the portable computing device 400 for display. IOS Interaction data 222 are generated and transmitted by the portable computing device 400 to the web server function 250, and forwarded to the first instance of IOS function IOS_2. The first instance of IOS function IOS_2 processes the IOS interaction data 222 and controls the execution of the first instance of Weather function executed by the server simulation functionalities 210, based on the processed IOS interaction data 222.

[0090] The second instance of IOS function IOS_2 transmits IOS control

and monitoring data 222 (e.g. a Weather control web page) to the portable computing device 400' for controlling the execution of the second instance of Weather function executed by the server simulation functionalities 210. The control data are transmitted by the second instance of IOS function IOS_2 to the web server function 250, and further transmitted to the portable computing device 400' for display. IOS Interaction data 222 are generated and transmitted by the portable computing device 400' to the web server function 250, and forwarded to the second instance of IOS function IOS_2. The second instance of IOS function IOS_2 processes the IOS interaction data 222 and controls the execution of the second instance of Weather function executed by the server simulation functionalities 210, based on the processed IOS interaction data 222. Although two instances of IOS function IOS_2 are mentioned in this example, a larger number of instances can be operating simultaneously on the simulation server 200.

[0091] Alternatively, the server simulation functionalities 210 may execute simultaneously an instance of Weather function in relation to the first simulation, and an instance of Navigation function in relation to the second simulation. Each instance of Weather and Navigation function executed by the server simulation functionalities 210 generates simulation data (e.g. a weather map and a navigation map), which are respectively transmitted directly to the simulators 300 and 300', via the web server function 250. The rendering functions 240 execute simultaneously an instance of Weather rendering function in relation to the first simulation, and an instance of Navigation rendering function in relation to the second simulation. Each instance of Weather and Navigation function executed by the server simulation functionalities 210 generates simulation data (e.g. the weather map and the navigation map), which are respectively adapted by the instances of Weather and Navigation rendering function, before transmission to the portable computing devices 400 and 400'.

[0092] In this case, the server IOS functionality 220 executes simultaneously an instance of IOS function IOS_1 and an instance of IOS function IOS_2, for respectively controlling the instances of Navigation and Weather function executed by the server simulation functionalities 210. The instances of IOS function IOS_1 and IOS function IOS_2 operate as previously described with respect to the portable computing devices 400 and 400'.

[0093] In a similar manner, the server GUI functionality 230 executes simultaneously a first instance of the server GUI functionality 230 in relation to the first simulation, and a second instance of the server GUI functionality 230 in relation to the second simulation. Each instance of the server GUI functionality 230 generates GUI graphical data 232, which are respectively transmitted to the simulators 300 and 300', via the web server function 250. Each instance of server GUI functionality 230 also receives, via the web server function 250, GUI interaction data 232 which are respectively transmitted by the simulators 300 and 300'. The simulator GUI 330 displays the GUI graphical data 232 received from the simulation server 200 on the simulators 300 and 300', and generates the GUI interaction data 232 based on the interactions of the users 20 with the displayed GUI graphical data 232. Each instance of the server GUI functionality 230 on the simulation server 200 processes the GUI interaction data 232 transmitted by the simulators 300 and 300', and respectively controls the execution of the first and second instance of the simulation, based on the processed GUI interaction data 232. Simulation data generated by the server simulation functionalities 210 are also transmitted directly by the web server function 250 (without adaptation by a rendering function 240) to the simulators 300 and 300', for display by the simulator GUI 330.

[0094] In an alternative embodiment, the web server function 250 is executed by a dedicated processing unit of the simulation server 200 (not represented in the Figures) different from the processing unit 201.

[0095] In another alternative embodiment, several instances of the web server function 250 are executed by the processing unit 201. For example, an instance is dedicated to the server IOS functionality 220, an instance is dedicated to the server simulation functionalities 210 and an instance is dedicated to the server GUI functionality 230. As mentioned previously, at least some of the instances may be executed by a dedicated processing unit different from the processing unit 201.

[0096] In still another alternative embodiment, the web server function 250 is not implemented on the simulation server 200, but on a standalone server not represented in the Figures.

[0097] Reference is now made concurrently to Figures 3A, 3B, 3C, 6A, 6B and 6C, where Figures 6A, 6B and 6C represent an exemplary flow diagram 600 illustrating interactions between the portable computing device 400 and components of the simulation server 200. More precisely, this exemplary flow diagram 600 illustrates interactions of the portable computing device IOS UI 420 with the simulation server 200.

[0098] At step 610, the user of the portable computing device 400 performs an authentication by entering its credentials, and the portable computing device web client function 450 transmits the credentials to the web server function 250. The web server function 250 verifies if the user is authorized to connect to the simulation portal based on the user credentials, and grants / denies access to the simulation portal based on the result of the verification of the user credentials. This step is optional, but is usually implemented to avoid that any user is granted access to the simulation portal without restrictions. An administrator of the simulation portal may be granted access to management functionalities of the portal, while standard users generally only have access to simulation functionalities of the portal.

[0099] At step 615, the portable computing device web client function

450 initiates a simulation web session with the web server function 250. For example, the user of the portable computing device 400 enters a Uniform Resource Locator (URL) corresponding to a simulation portal hosted by the web server function 250, and the portable computer device web client function 450 requests a connection to the simulation server 200, hereinafter referred as the simulation portal. In return, the web server function 250 returns a home page of the simulation portal to be displayed by the portable computing device web client function 450.

[00100] At step 620, the web server function 250 transmits a list of candidate server simulation functionalities 210 (e.g. Weather function, Navigation function, etc.) to the portable computing device web client function 450. The list may be determined based on a particular profile of the user, and may comprise only a subset (e.g. Weather function only) of all available server simulation functionalities 210 supported by the web server function 250. The subset corresponds to server simulation functionalities 210 (e.g. Weather function only) that the user of the portable computing device 400 is authorized to use based on its profile. For each user, the web server function 250 stores a profile of the user for determining the corresponding authorized server simulation functionalities 210. The profile of each user can be generated by an administrator of the web server function 250. For example, in the case of an aircraft simulation, the user may only be authorized to use server simulation functionalities 210 corresponding to one or more particular type(s) of aircraft, to one or more particular system(s) or sub-system(s) of an aircraft, to military or civilian aircrafts only, etc. The portable computing device web client function 450 displays the list of candidate server simulation functionalities 210 (e.g. Weather function and Navigation function) for allowing the user to select one among the list of candidates. The selection of a particular server simulation functionality (e.g. Weather function) in the list of candidate server simulation functionalities 210 by the user is transmitted to the

web server function 250 function by the portable computing device web client function 450. Alternatively, a plurality of candidate server simulation functionalities 210 can be selected simultaneously.

[00101] At step 621, the web server function 250 determines if an instance of the selected server simulation functionality 210 (e.g. Weather function) is already running, and if it is not the case, launches such an instance. Since the simulation server 200 can support a plurality of simulation sessions in parallel, a server simulation functionality 210 (e.g. Weather function) may have several instances running in parallel for different independent simulation sessions. Thus, before executing step 620, the user may need to join an existing simulation session or create a new simulation session. The creation of / joining to a simulation session is performed through interactions between the portable computing device web client function 450 and the web server function 250. Each independent simulation session is allocated a unique identifier and information describing the characteristics of the simulation session, so that any portable computing device 400 or simulator 300 can join the proper ongoing simulation session based on its characteristics (e.g. type of aircraft simulated, simulator 300 used for the simulation, etc.).

[00102] The selected server simulation functionality 210 (e.g. Weather function) may automatically provide access to corresponding IOS function(s) of the server IOS functionality 220 (e.g. IOS function IOS_2). Alternatively, an interactive selection step 622 similar to selection step 620 is performed. At step 622, the web server function 250 transmits a list of candidate IOS function(s) of the server IOS functionality 220 (e.g. IOS function IOS_1, IOS function IOS_2, etc.) to the portable computing device web client function 450. The list may be determined based on a particular profile of the user, and may comprise only a subset (e.g. IOS function IOS_2 only) of all available server IOS functions supported by the web server function 250. The subset corresponds to server IOS

functions 220 (e.g. IOS function IOS_2 only) that the user of the portable computing device 400 is authorized to use based on its profile. For each potential user, the web server function 250 stores a profile of the user for determining the corresponding authorized server IOS functions 220. The portable computing device web client function 450 displays the list of candidate server IOS functions 220 (e.g. IOS function IOS_1 and IOS function IOS_2) for allowing the user to select one among the list of candidates. The selection of server IOS functions 220 (e.g. IOS function IOS_2) in the list of candidate server IOS functions by the user is transmitted to the web server function 250 by the portable computing device web client function 450. Alternatively, a plurality of candidate server IOS functions 220 can be selected simultaneously.

[00103] At step 625, the web server function 250 launches an instance of server IOS function 220 corresponding to the IOS function (e.g. IOS_2) selected at step 622. The web server function 250 also launches an instance of server rendering function 240 (e.g. Weather rendering function) corresponding to the server simulation functionality 210 (e.g. Weather function) selected at step 620. As mentioned previously, several instances of the same server IOS function 210 or server rendering function 240 can be executed in parallel by the simulation server 200, for supporting a plurality of simulation sessions running in parallel, and also for supporting a plurality of portable computing devices 400 participating in parallel to the same or similar simulation sessions. Furthermore, a single portable computing device 400 may be interacting with a plurality of server IOS functions 210 and / or a plurality of server rendering functions 240 in parallel.

[00104] The web server function 250 establishes a first communication channel for exchanging data between the portable computing device 400 and the instance of server IOS function 220 (e.g. IOS function IOS_2) launched at step 625; and optionally between the portable computing device 400 and the instance of server simulation functionality 210 (e.g. Weather function) launched at step 621

(to directly transmit simulation data which do not need to be adapted by a rendering function). The data are exchanged between the simulation server 200 and the portable computing device 400 through the web server function 250 and the portable computing device web client 450. All the data exchanged through this first communication channel do not need to be adapted for rendering on the portable computing device 400 via a server rendering function 240.

[00105] The web server function 250 establishes a second communication channel between the instance of server rendering function 240 (e.g. Weather rendering function) launched at step 625 and the portable computing device 400, for transmitting simulation data adapted for rendering on the portable computing device 400. Establishing this second communication channel is well known in the art, and may comprise determining a connection identification, selecting communication protocol(s), allocating communication sockets, etc.

[00106] The web server function 250 may create and manage a dynamic communication profile for each portable computing device 400, comprising characteristics of the created first and second communication channels. The management of the dynamic communication profile includes creation / update / deletion of the first and second communication channels.

[00107] Furthermore, the web server function 250 provides the launched instance of server rendering function 240 with characteristics of the portable computing device 400. The characteristics include for example processing power, memory size, display resolution, data throughput of a communication interface, available user interfaces, etc. These characteristics are used by the launched instance of server rendering function 240 for performing the adaptation of the simulation data transmitted to the portable computing device 400. For each authorized user of the simulation portal, the web server function 250 may store a static profile (with the aforementioned characteristics) of the portable computing

device 400 used by the user. Alternatively, the web server function 250 automatically generates a dynamic profile (with the aforementioned characteristics) of the portable computing device 400 used by the user at step 610, by dynamically retrieving the characteristics of the device 400 currently used by the user (this procedure is well known in the art of web browsing).

[00108] Figure 6B more specifically represents the transmission of adapted simulation data by the instance of server rendering function 240 to the portable computing device 400.

[00109] At step 630, the instance of server simulation functionality 210 (e.g. Weather function) generates simulation data and forwards the simulation data to the corresponding instance of server rendering function 240 (e.g. Weather rendering function).

[00110] At step 635, the instance of server rendering function 240 processes the simulation data, and generates simulation data adapted (based on the aforementioned characteristics of the portable computing device 400) for rendering on the portable computing device 400.

[00111] At step 640, the adapted simulation data are transmitted directly by the instance of server rendering function 240 to the portable computing device 400 (without using the web server function 250).

[00112] At step 645, the display function 421 of the portable computing device IOS UI 420 processes the adapted simulation data received from the instance of server rendering function 240, and displays the processed simulation data on the display 404 of the portable computing device 400. Since the simulation data have been adapted to the device 400 at step 635, the processing is very limited and may even not be needed before displaying the simulation data.

[00113] Although a single sequence of steps 630, 635, 640 and 645 is represented in Figure 6B for simplification purposes, a plurality of sequences may

occur. For each sequence, simulation data adapted for rendering on the portable computing device 400 are generated at steps 630 and 635, transmitted at step 640 and displayed at step 645.

[00114] Figure 6C more specifically represents the exchange of data not adapted by a server rendering function 240 between the web server function 250 and the portable computing device 400.

[00115] At step 650, the instance of server IOS function 220 (e.g. IOS_2) generates IOS control and monitoring data 222 (not adapted by a rendering function), and forwards the IOS control and monitoring data 222 to the web server function 250.

[00116] At step 655, the IOS control and monitoring data 222 are transmitted by the web server function 250 to the portable computing device 400 (without applying any rendering function).

[00117] At step 660, the display function 421 of the portable computing device IOS UI 420 displays the received IOS control and monitoring data 222 on the display 404 of the portable computing device 400.

[00118] At step 665, the interaction function 422 of the portable computing device IOS UI 420 generates IOS interaction data 222 based on interactions of the user of the portable computing device 400 (e.g. with the IOS control and monitoring data 222 displayed at step 660).

[00119] At step 670, the IOS interaction data 222 are transmitted by the portable computing device 400 to the web server function 250. The web server function 250 simply forwards the IOS interaction data 222 to the instance of server IOS function 220 (e.g. IOS_2).

[00120] The web server function 250 may implement a filtering function (not represented in the Figures), for identifying and adequately handling the data received from the portable computing device(s) 400. The filtering function

identifies IOS interaction data 222 received at step 670, which shall be forwarded to the proper instance of server IOS function 220 (e.g. IOS_2). The filtering function also identifies administrative and management data received at steps 610, 615, 620 and 622 of Figure 6A, which shall be processed directly by the web server function 250.

[00121] At step 675, the instance of server IOS function 220 (e.g. IOS_2) processes the IOS interaction data 222 to control the execution of the simulation executed by the simulation server 200 based on the IOS interaction data 222.

[00122] Although a single sequence of steps 650, 655 and 660 is represented in Figure 6C for simplification purposes, a plurality of sequences may occur. Similarly, a plurality of sequences of steps 665, 670 and 675 may occur. A plurality of sequences of steps 650, 655 and 660 may occur before a single sequence of steps 665, 670 and 675 occurs. Similarly, a plurality of sequences of steps 665, 670 and 675 may occur before a single sequence of steps 650, 655 and 660 occurs. However, a sequence of steps 665, 670 and 675 is generally followed by a sequence of steps 650, 655 and 660 (and / or steps 630, 635, 640 and 645 of Figure 6B); since the processing of the IOS interaction data 222 impacts the execution of the simulation executed by the simulation server 200, which in turn leads to the generation of new adapted simulation data 212 / IOS control and monitoring data 222 which are transmitted to the portable computing device 400.

[00123] Although not represented in Figure 6C for simplification purposes, steps 650, 655 and 660 may also consist in the generation by the instance of server simulation functionality 210 represented in Figure 6B of simulation data which do not need to be adapted by a rendering function, the transmission of these simulation data to the portable computing device 400 directly by the web server function 250, and the display of these simulation data on the portable computing device 400

[00124] Furthermore, the transmission of simulation data adapted by the instance of server rendering function 240 as illustrated in Figure 6B, and the transmission of IOS control and monitoring data 222 / simulation data 212 by the web server function 250 (without adaptation by a rendering function) as illustrated in Figure 6C, occur simultaneously and independently.

[00125] As is well known in the art, the communications between the web server function 250 and the portable computing device web client 450 use the Hypertext Transfer Protocol (HTTP) and / or Hypertext Transfer Protocol Secure (HTTPS). Optionally, the Real-time Transport Protocol (RTP) may also be used for some of the data exchanged between the web server function 250 and portable computing device(s) 400. A single step represented in Figures 6A and 6C (e.g. 610, 615, 620, 622, 655 and 670) may include a plurality of HTTP / HTTPS / RTP messages exchanged between the web server function 250 and portable computing device(s) 400.

[00126] Similarly, the communications between the instances of server rendering function 240 and portable computing device(s) 400 may also use the HTTP and / or HTTPS and / or RTP protocols. A single step represented in Figure 6B (e.g. 640) may include a plurality of HTTP / HTTPS / RTP messages exchanged between the instances of server rendering function 240 and portable computing device(s) 200. In this case, each instance of server rendering function 240 implements an autonomous HTTP based server allowing communications with the portable computing device(s) 400 via web sockets. The establishment of the communication channel between the instances of server rendering function 240 and portable computing device(s) 400 is performed under the direction of the web server function 250 at step 625. However, this communication channel is not limited to the use of the HTTP and / or HTTPS and / or RTP protocols, but may use other non-web-based communication protocols (e.g. a proprietary communication protocol).

[00127] The flow diagram 600 is for illustration purposes only. A similar flow diagram may be adapted to illustrate interactions of the portable computing device GUI 430 represented in Figure 2B, with the simulation server 200. Additionally, similar flow diagrams may be respectively adapted to illustrate interactions of the simulator IOS UI 320, simulator GUI 330 and simulator simulation functionalities 310 executed on the simulator 300 represented in Figure 2A, with the simulation server 200.

[00128] Reference is now made concurrently to Figures 3A, 3B, 3C and 5. Figure 5 represents an IOS UI page 500 displayed on the display 304 of the simulator 300. The IOS UI page 500 is displayed by the display function 321 of the simulator IOS UI 320. The IOS UI page 500 provides a user interface with sub-groups of instances. The IOS UI page 500 generally includes graphical control elements (e.g. menus and sub-menus, list boxes, etc.) for controlling simulation parameters, and graphical display elements (e.g. images, text fields, icons, embedded videos, etc.) for displaying simulation data generated by the server simulation functionalities 210.

[00129] The IOS UI page 500 represented in Figure 5 comprises a first image 501 (Navigation map), a graphical control element 502 (control widget), and a second image 503 (Weather map). The Navigation map is generated by the Navigation function of the server simulation functionalities 210, and directly transmitted (without adaptation by the server rendering function 240) to the simulator IOS UI 320 via the web server function 250, for display on the simulator display 304 by the simulator display function 321. The Navigation map is updated based on the execution of the Navigation function of the server simulation functionalities 210. The Weather map 503 is generated by the Weather function of the server simulation functionalities 210, and directly transmitted (without adaptation by the server rendering function 240) to the simulator IOS UI 320 via the web server function 250, for display on the simulator display 304 by the

simulator display function 321. The Weather map is updated based on the execution of the Weather function of the server simulation functionalities 210.

[00130] The web server function 250 receives IOS control and monitoring data 222 corresponding to the control widget 502 (allowing control of the Navigation map 501 and Weather map 503) from the server IOS functionality 220. The IOS control and monitoring data 222 are transmitted to the simulator 300 by the web server function 250, and the control widget 502 is displayed on the simulator display 304 by the simulator display function 321 of the simulator IOS UI 321 based on the received IOS control and monitoring data 222. The control widget 502 is used by a user for modifying parameters related to the Navigation map 501 and the Weather map 503, when the user interacts with the simulator 300 via the interaction function 322 of the simulator IOS UI 320. IOS Interaction data 222 comprising the modified parameters are transmitted by the interaction function 322 of the simulator IOS UI 320 to the server IOS functionality 220, for controlling the execution of the Navigation and Weather functions of the server simulation functionalities 210.

[00131] Figure 5 also represents an IOS UI page 510 displayed on the display 404 of the portable computing device 400. The IOS UI page 510 is displayed by the portable computing device display function 421 of the portable computing device IOS UI 420. The IOS UI page 510 comprises an image 511 (Navigation map) corresponding to the Navigation map 501 of the IOS UI page 500, and a graphical control element 512 (control widget) corresponding to the control widget 502 of the IOS UI page 500.

[00132] For illustration purposes, the user of the portable computing device 400 has decided not to use the Weather function of the server simulation functionalities 210, and consequently an image corresponding to the Weather map 503 of the IOS UI page 500 is not displayed on the display 404 of the portable computing device 400. In an alternative use case not represented in

Figure 5, if the user of the portable computing device 400 had decided to use the Weather function of the server simulation functionalities 210, an image corresponding to the Weather map 503 of the IOS UI page 500 would be displayed on the portable computing device display 404.

[00133] On the simulation server 200, an instance of Navigation rendering function 240 receives simulation data corresponding to the Navigation map 511 from the Navigation function of the server simulation functionalities 210. The instance of Navigation rendering function 240 processes the simulation data to generate the Navigation map 511 adapted for rendering on the portable computing device 400. For example, the size and resolution of the Navigation map 511 is adapted to characteristics (e.g. screen resolution, etc.) of the portable computing device 400. The Navigation map 511 is transmitted to the portable computing device 400 by the instance of server rendering function 240, and displayed on the portable computing device display 404 by the portable computer device IOS UI 421.

[00134] The web server function 250 at the simulation server 200 receives IOS control and monitoring data 222 corresponding to the control widget 512 (allowing control of the Navigation map 511) from the server IOS functionality 220. The IOS control and monitoring data 222 are transmitted to the portable computing device 400 by the web server function 250, and the control widget 512 is displayed on the portable computing device display 404 by the portable computing device IOS UI 421 based on the received IOS control and monitoring data 222.

[00135] When the user 10 interacts with the IOS UI page 510 via a user interface 405 of the portable computing device 400, corresponding IOS interaction data 222 are generated by the interaction function 422 of the portable computing device IOS UI 420, and transmitted by the interaction function 422 to the web server function 250 of the simulation server 200. The web server function 250

forwards the IOS interaction data 222 to the server IOS functionality 220.

[00136] For example, the control widget 512 is a menu comprising three items. When the user positions a pointer (corresponding to a mouse) on one of the items and left clicks, the transmitted IOS interaction data 222 comprise the selected item.

[00137] Alternatively or complementarity, the user 10 may interact directly with an area of the IOS UI page 510 without using the control widget 512. For example, the user 10 may position a pointer (corresponding to a mouse) on the Navigation map 511, and left click or right click on the Navigation map 511. The transmitted IOS interaction data 222 comprise an indication that the user 10 interacted with the Navigation map 511, and more specifically via a right-click or a left-click. The IOS interaction data 222 are interpreted by the server IOS functionality 220 at the simulation server 200 as follows: a left-click is a zoom-in request and a right-click is a zoom-out request. The server IOS functionality 220 reconfigures the Navigation function of the server simulation functionalities 210 accordingly. In case of a zoom-in, the Navigation function of the server simulation functionalities 210 generates more detailed simulation data, which are processed by the instance of Navigation server rendering function 240 for generating a zoomed-in Navigation map 511 for rendering on the portable computing device 400. In case of a zoom-out, the Navigation function of the server simulation functionalities 210 generates less detailed simulation data, which are processed by the instance of Navigation server rendering function 240 for generating a zoomed-out Navigation map 511 for rendering on the portable computing device 400.

[00138] More generally, the IOS interaction data 222 are used by the server IOS functionality 220 for controlling the corresponding server simulation functionality 210 (e.g. Navigation function). Controlling the corresponding server simulation functionality 210 includes controlling the simulation data generated by

the server simulation functionality 210 (e.g. Navigation function), which are further adapted by the corresponding instance of server rendering function 240 (e.g. instance of Navigation server rendering function 240) for rendering (e.g. Navigation map 511) on the portable computing device display 404.

[00139] The web server function 250 may pre-process the received IOS interaction data 222 to determine if they correspond to a legitimate interaction with the IOS UI page 510 displayed on the portable computing device 400. The web server function 250 simply discards transmitted IOS interaction data 222 which do not correspond to a legitimate interaction with the IOS UI page 500, and transmits legitimate interactions to the server IOS functionality 220. The web server function 250 may further discriminate the IOS interaction data 222 generated with the IOS UI page 510, from other types of data (e.g. administration and configuration of the simulation server 200) which are processed directly by the web server function 250.

[00140] Although the present disclosure has been described hereinabove by way of non-restrictive, illustrative embodiments thereof, these embodiments may be modified at will within the scope of the appended claims without departing from the spirit and nature of the present disclosure.

WHAT IS CLAIMED IS:

1. A simulation server comprising:
 - a communication interface for exchanging data with other entities;
 - a processing unit for:
 - executing at least one simulation;
 - generating simulator simulation data and transmitting the simulator simulation data to at least one simulator via the communication interface, the simulator simulation data being representative of the execution of the at least one simulation;
 - receiving simulator interaction data from the at least one simulator via the communication interface; and
 - processing the simulator interaction data and controlling the execution of the at least one simulation based on the processed simulator interaction data.
2. The simulation server of claim 1, wherein:
 - the processing unit executes a plurality of simulations; and
 - for each specific simulation among the plurality of simulations, the processing unit:
 - generates simulator simulation data and transmits the simulator simulation data to a specific simulator,
 - receives simulator interaction data from the specific simulator,
 - processes the simulator interaction data, and

controls the execution of the specific simulation based on the processed simulator interaction data.

3. The simulation server of claim 1, wherein the processing unit further generates and transmits simulator Instructor Operating Station (IOS) control and monitoring data to a specific simulator among the at least one simulator, the simulator IOS control and monitoring data allowing the generation of the simulator interaction data by the specific simulator via interactions of a user with the specific simulator when the simulator IOS control and monitoring data are displayed on the specific simulator.
4. The simulation server of claim 1, wherein the processing unit further generates and transmits simulator graphical data to a specific simulator among the at least one simulator, the simulator graphical data allowing the generation of the simulator interaction data by the specific simulator via interactions of a user with the specific simulator when the simulator graphical data are displayed on the specific simulator.
5. The simulation server of claim 1, wherein the processing unit further:
 - generates portable computing device simulation data and transmits the portable computing device simulation data to at least one portable computing device via the communication interface, the portable computing device simulation data being representative of the execution of the at least one simulation;
 - receives portable computing device interaction data from the at least one portable computing device via the communication interface; and
 - processes the portable computing device interaction data and controls the execution of the at least one simulation based on the processed portable computing device interaction data.
6. The simulation server of claim 5, wherein:

the processing unit executes a plurality of simulations; and

for each specific simulation among the plurality of simulations, the processing unit:

generates portable computing device simulation data and transmits the portable computing device simulation data to a specific portable computing device,

receives portable computing device interaction data from the specific portable computing device,

processes the portable computing device interaction data, and

controls the execution of the specific simulation based on the processed portable computing device interaction data.

7. The simulation server of claim 5, wherein the processing unit further executes at least one rendering function, the rendering function processing at least some of the portable computing device simulation data transmitted to a specific portable computing device to generate portable computing device simulation data adapted for rendering on the specific portable computing device.
8. The simulation server of claim 7, wherein the rendering function uses characteristics of the portable computing device for generating the portable computing device simulation data adapted for rendering on the specific portable computing device, the characteristics including at least one of the following: processing power, memory size, display resolution, data throughput of a communication interface, and available user interfaces.
9. The simulation server of claim 7, wherein the portable computing device simulation data comprise at least one simulation image and generating

portable computing device simulation data adapted for rendering on the specific portable computing device comprises at least one of the following: encoding the simulation image in a particular format, applying a particular compression algorithm to the simulation image, applying a particular image sampling algorithm to the simulation image, and applying a particular algorithm for lowering an image resolution of the simulation image.

10. The simulation server of claim 5, wherein the processing unit further generates and transmits portable computing device IOS control and monitoring data to a specific portable computing device among the at least one portable computing device, the portable computing device IOS control and monitoring data allowing the generation of the portable computing device interaction data by the specific portable computing device via interactions of a user with the specific portable computing device when the portable computing device IOS control and monitoring data are displayed on the specific portable computing device.
11. The simulation server of claim 5, wherein the processing unit further generates and transmits portable computing device graphical data to a specific portable computing device among the at least one portable computing device, the portable computing device graphical data allowing the generation of the portable computing device interaction data by the specific portable computing device via interactions of a user with the specific portable computing device when the portable computing device graphical data are displayed on the specific portable computing device.
12. The simulation server of claim 1, wherein for each of the at least one simulation, executing the simulation by the processing unit comprises executing at least one simulation functionality corresponding to a specific simulator by the processing unit, and the simulator simulation data generated by the execution of the at least one simulation functionality are

transmitted to the specific simulator.

13. The simulation server of claim 1, wherein the simulation data comprise at least one of the following: a static simulation image and a flow of simulation images.
14. The simulator of claim 1, wherein the simulation data comprise a static simulation image consisting in a static map.
15. The simulator of claim 1, wherein the simulation data comprise a flow of simulation images consisting in a real time streaming of a map.
16. The simulation server of claim 1, wherein the communication interface supports at least one of the following communication protocols: Wi-Fi, and Ethernet.
17. The simulation server of claim 1, wherein the simulation consists in a simulation of an aircraft.

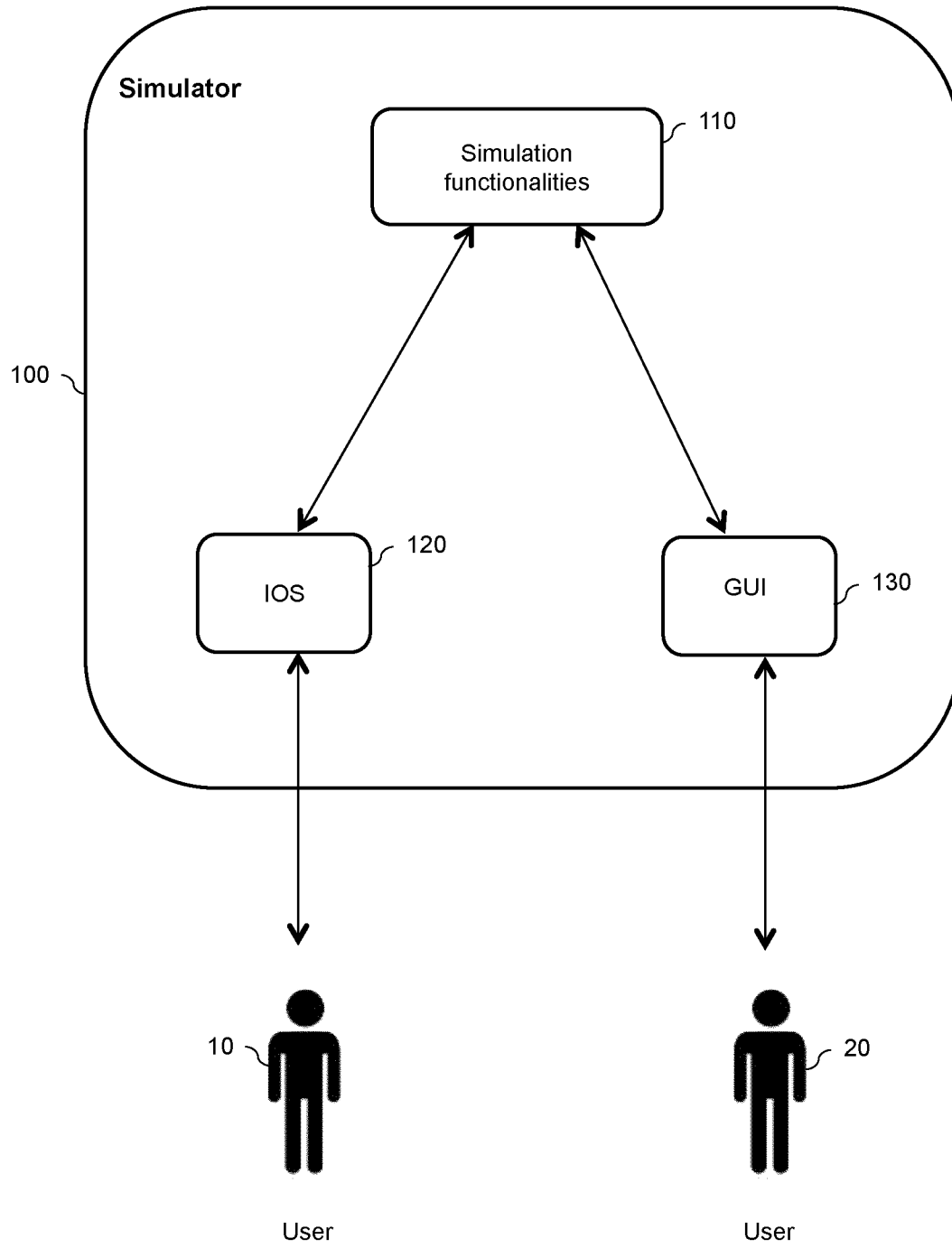


Figure 1

2/17

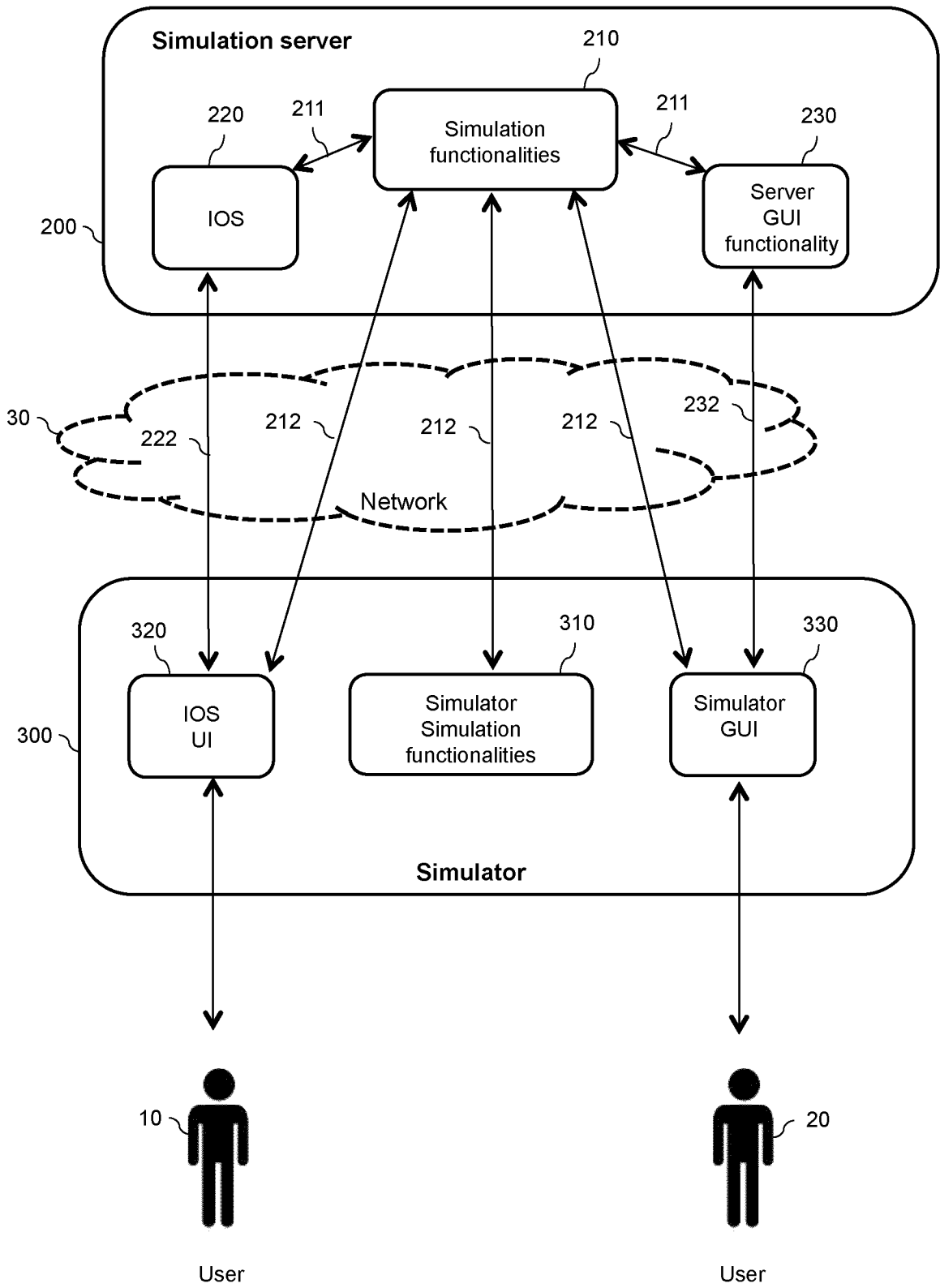


Figure 2A

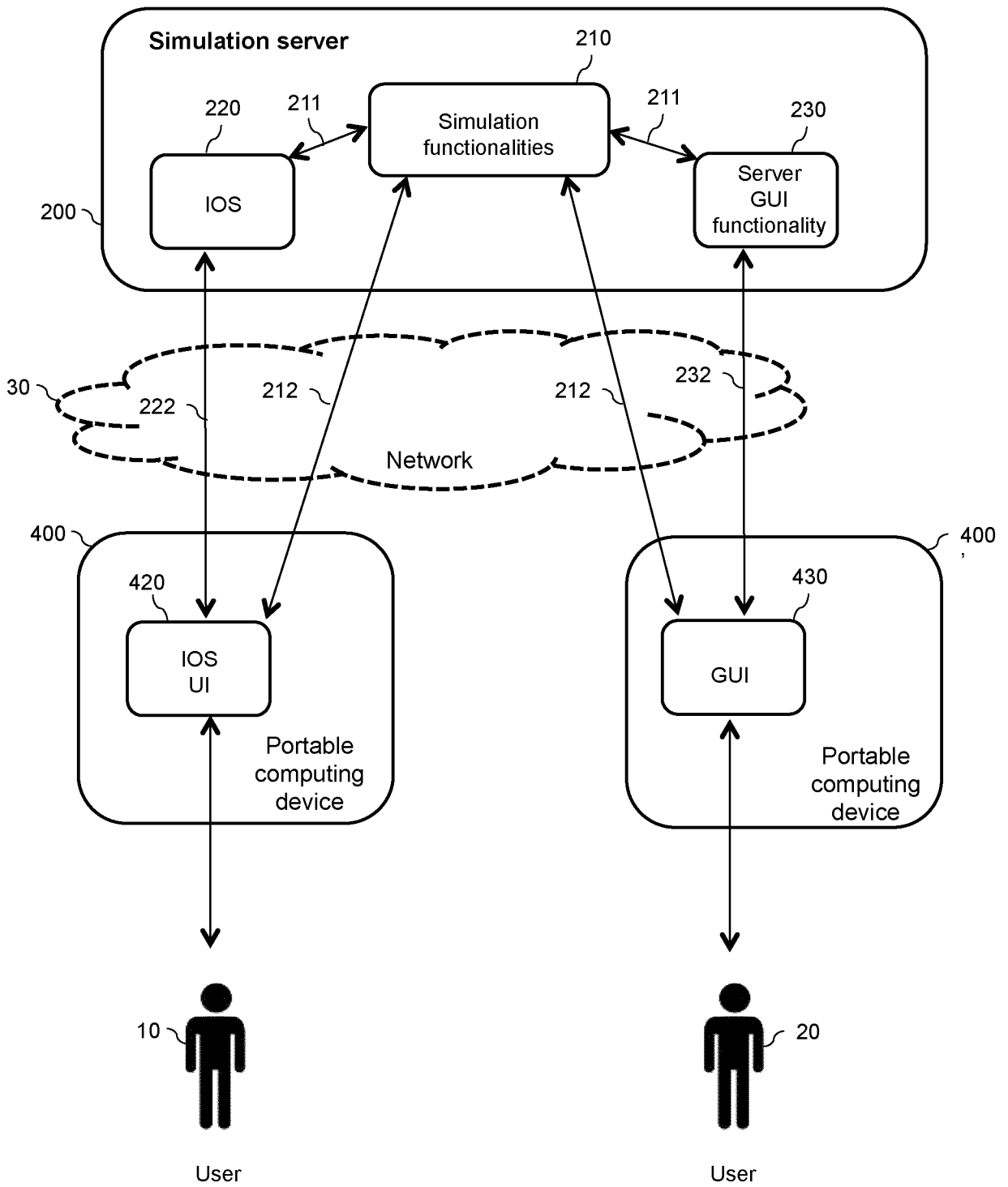


Figure 2B

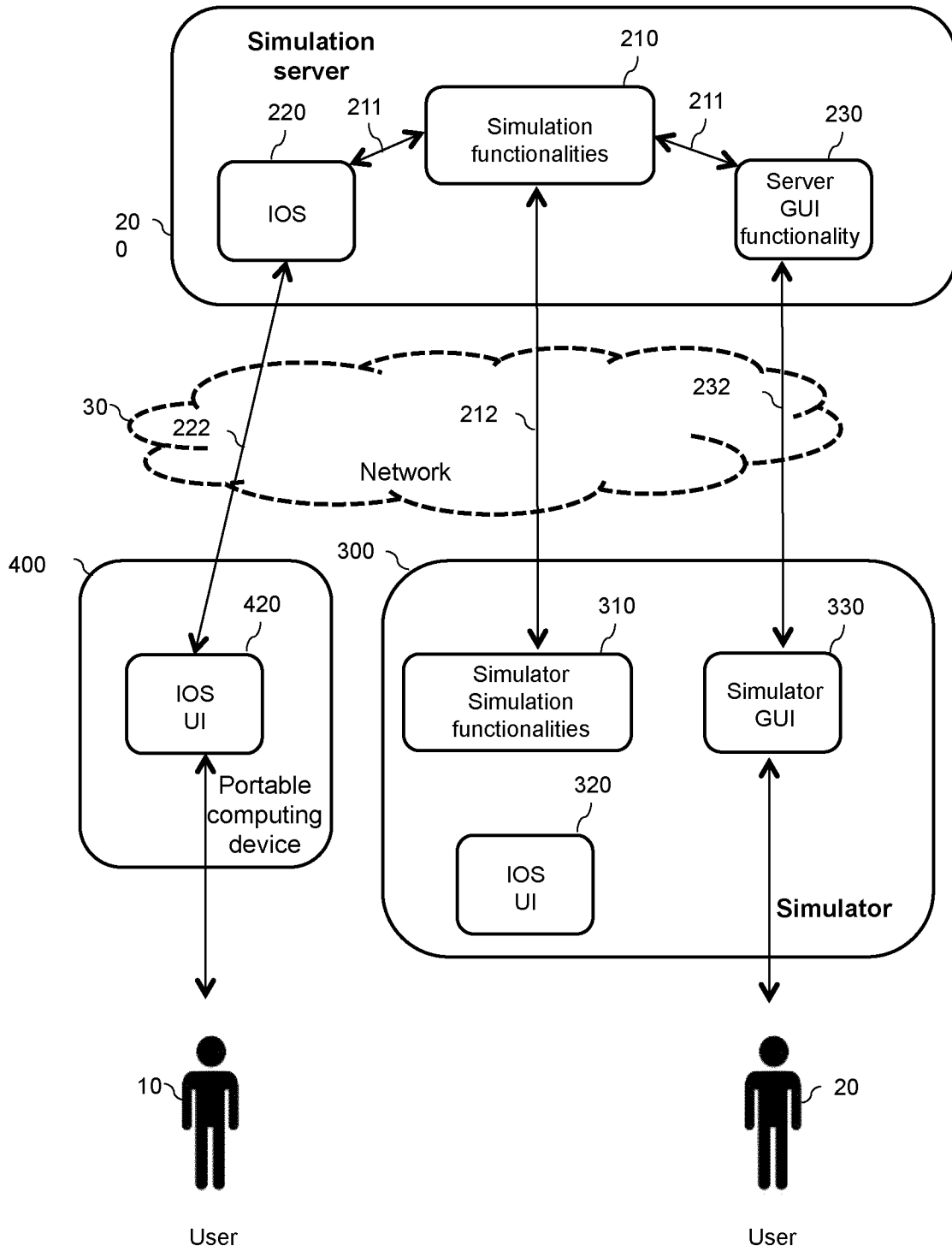


Figure 2C

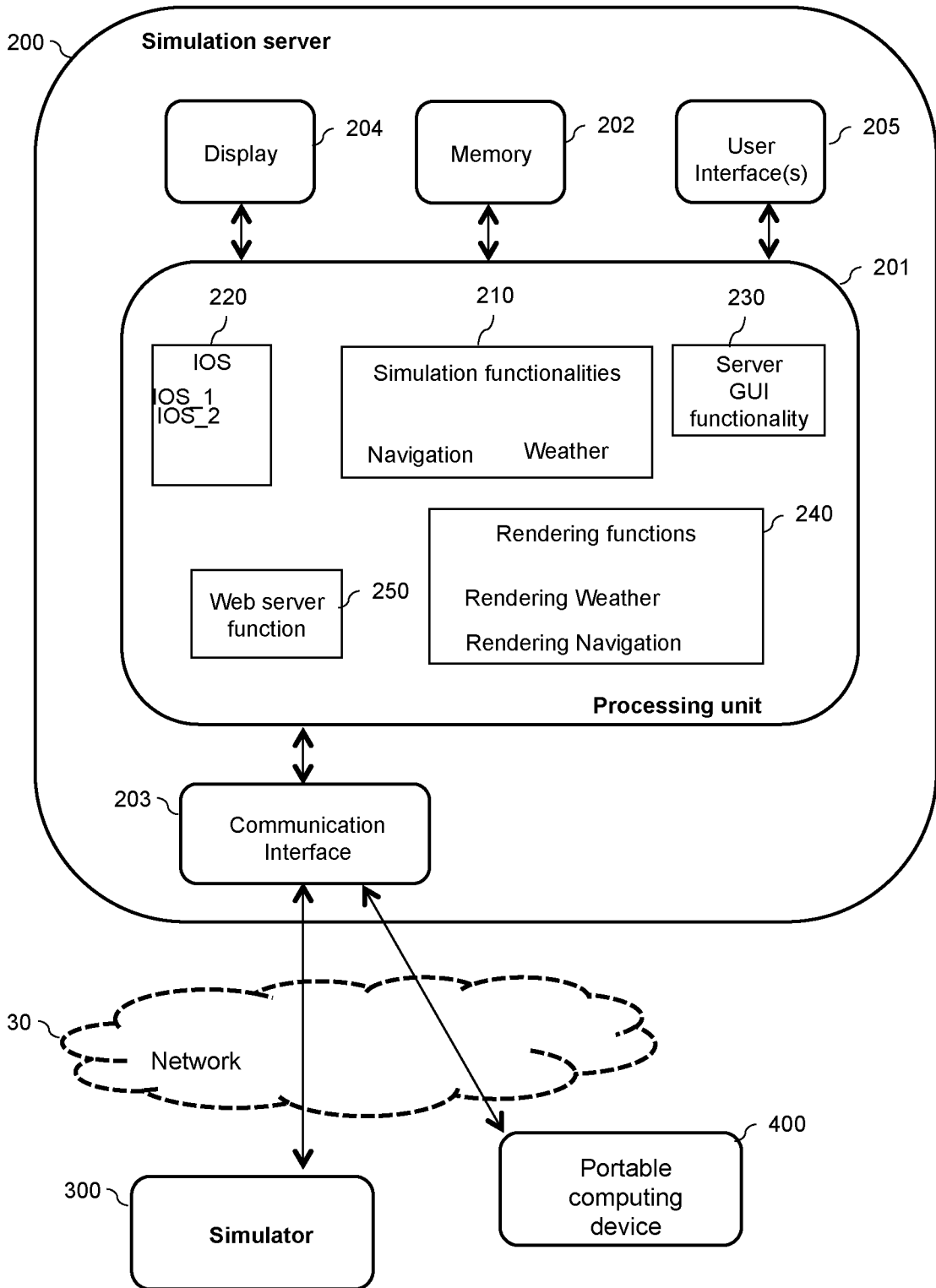


Figure 3A

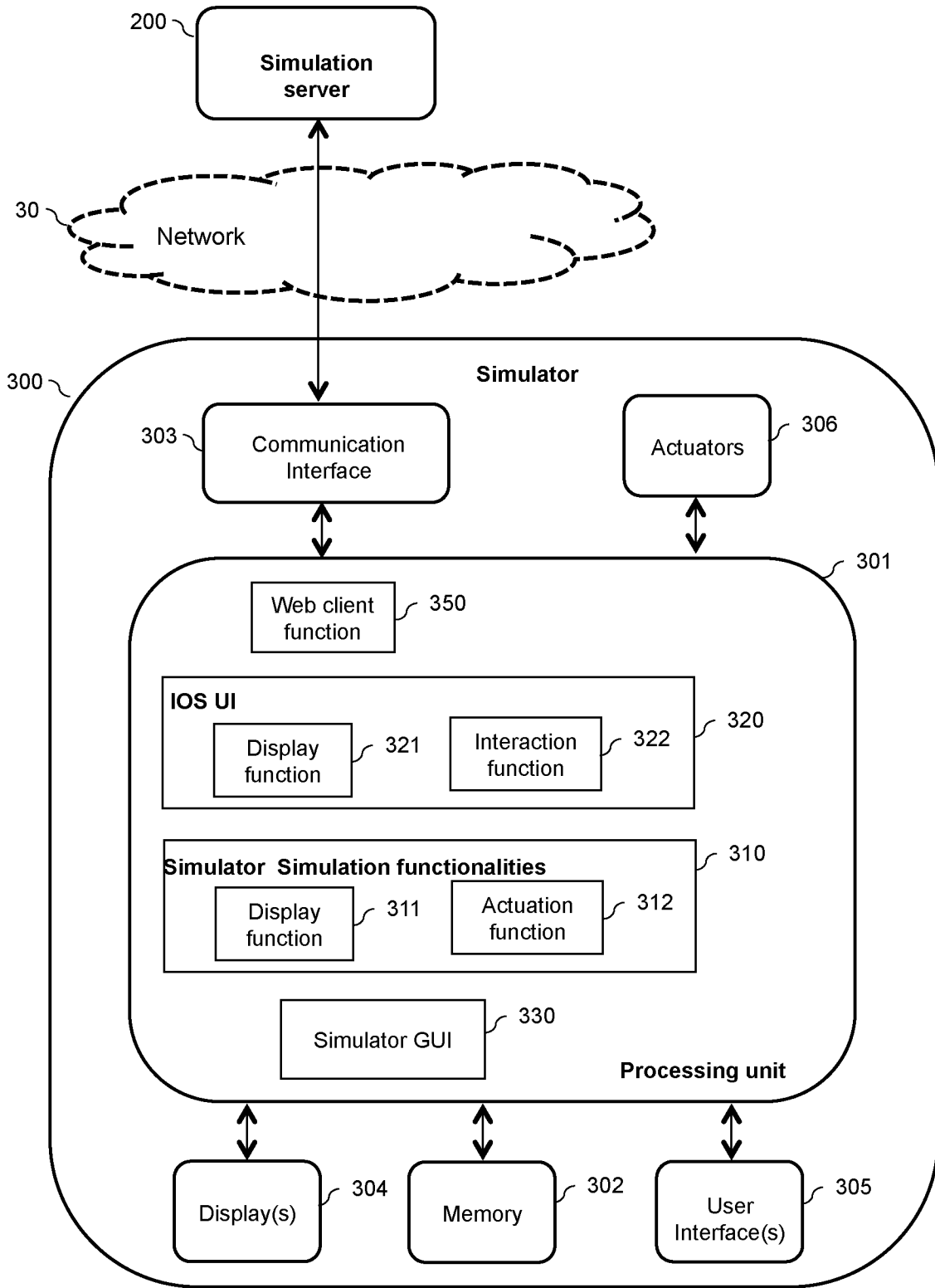


Figure 3B

7/17

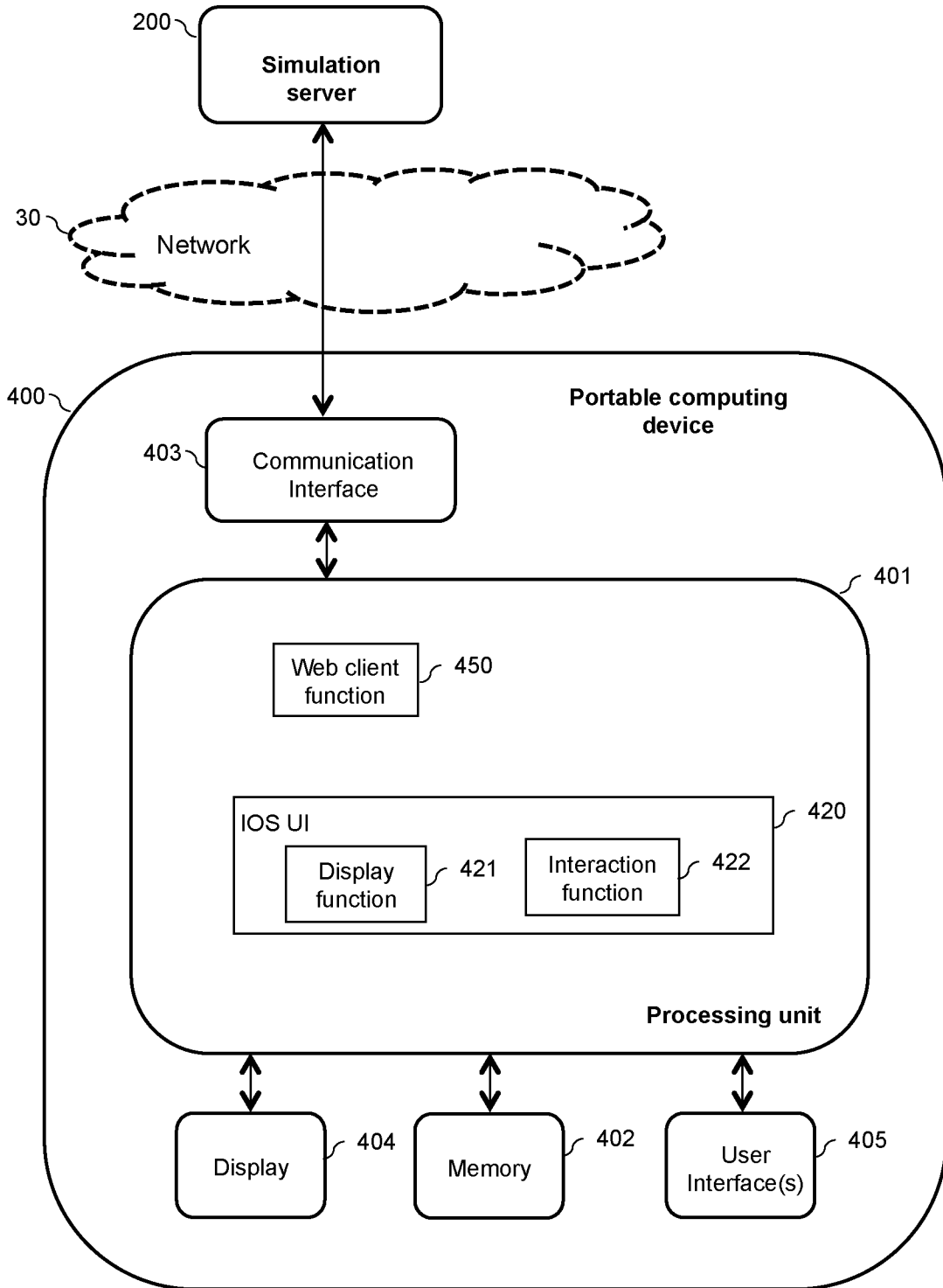


Figure 3C

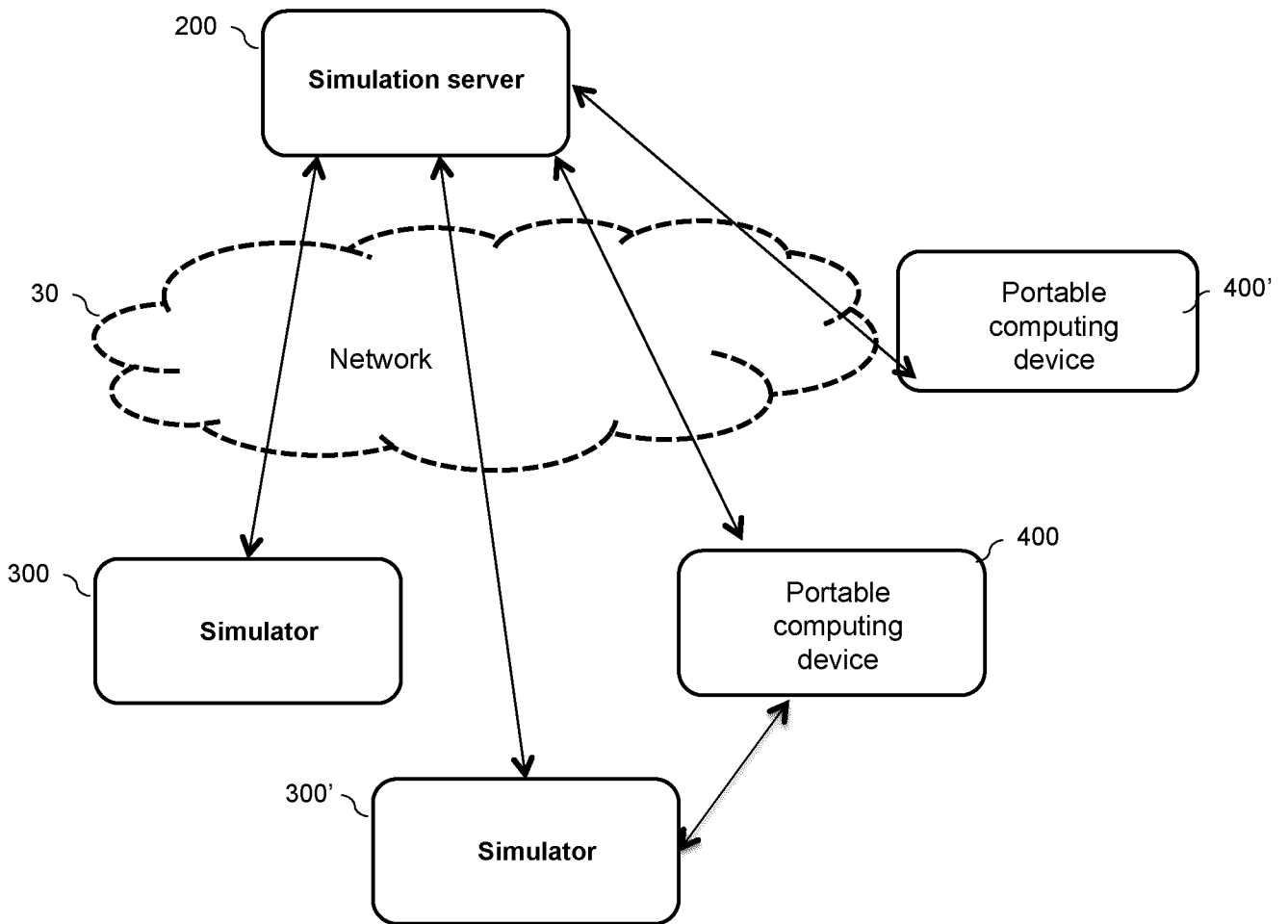


Figure 4

9/17

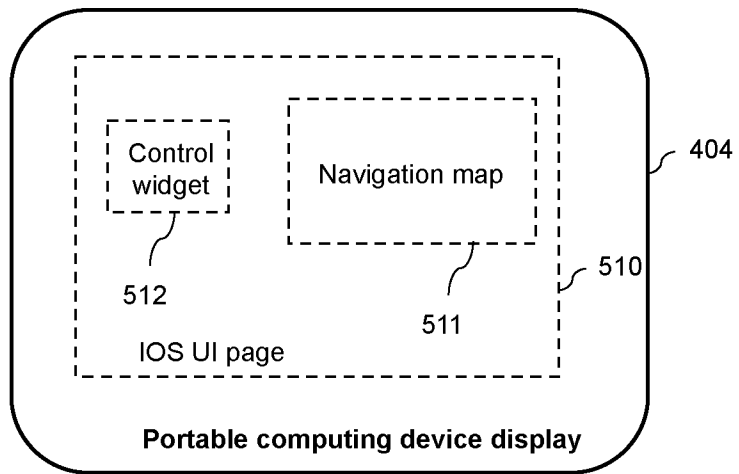
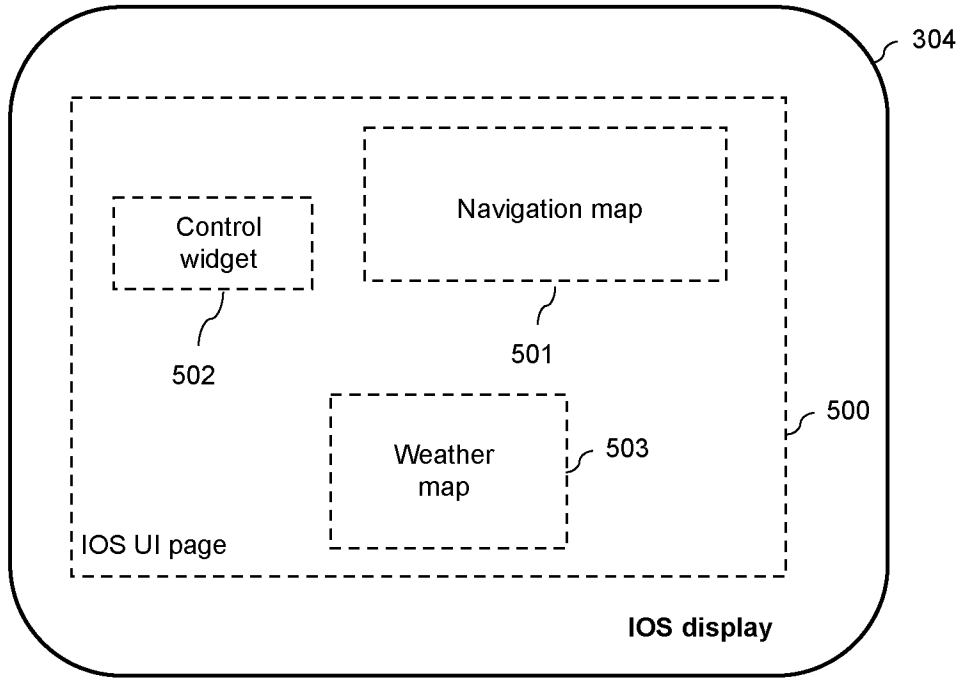


Figure 5

10/17

600

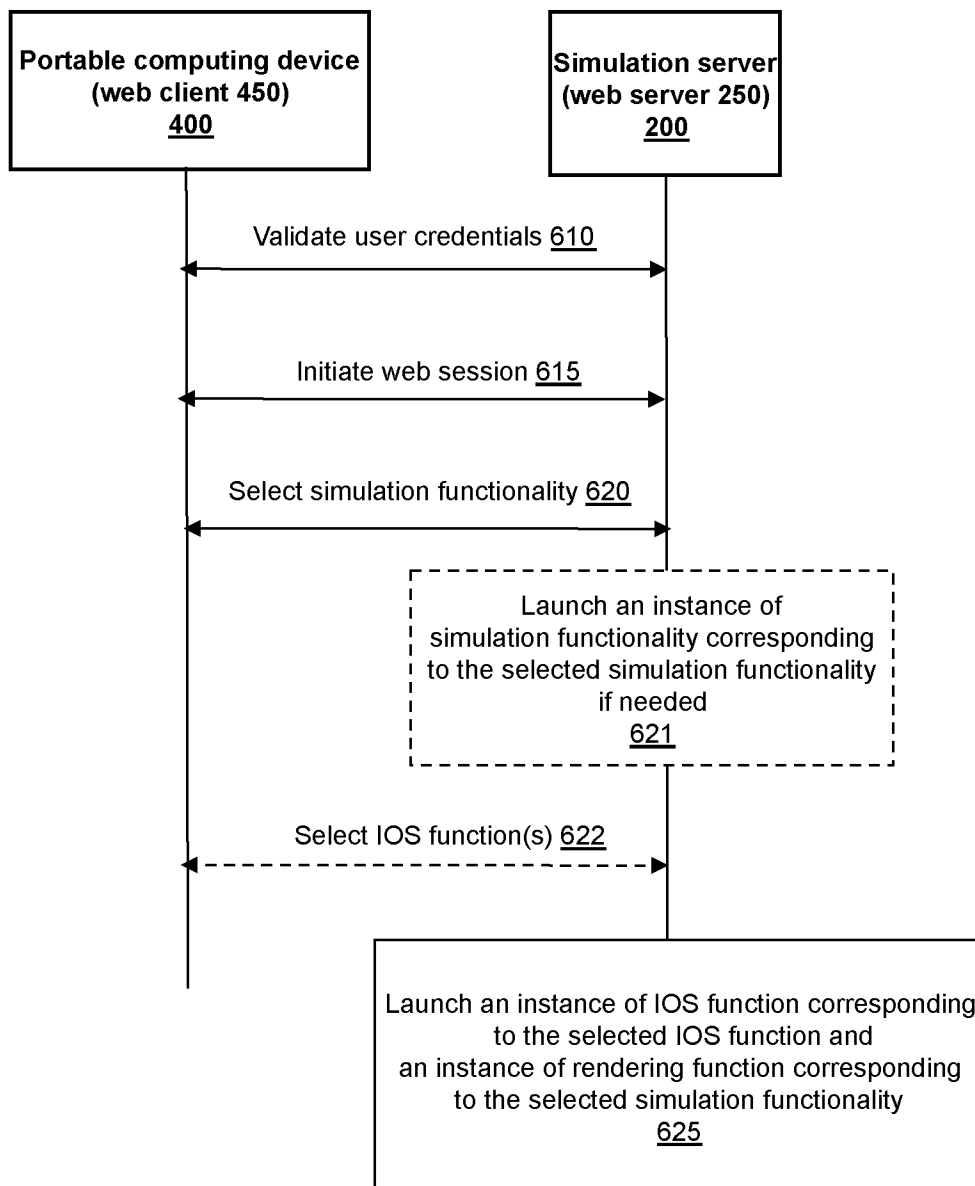


Figure 6A

11/17

600

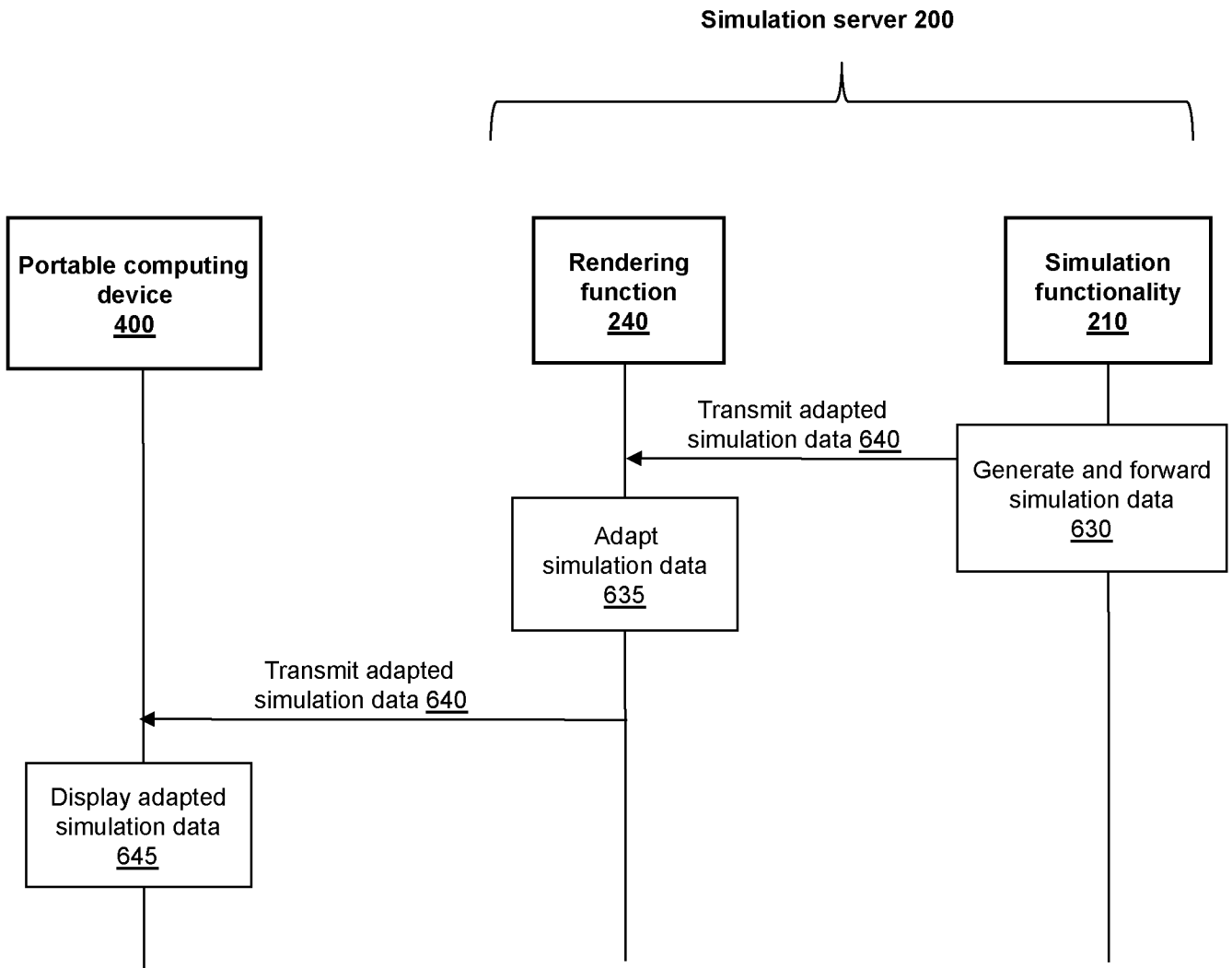


Figure 6B

12/17

600

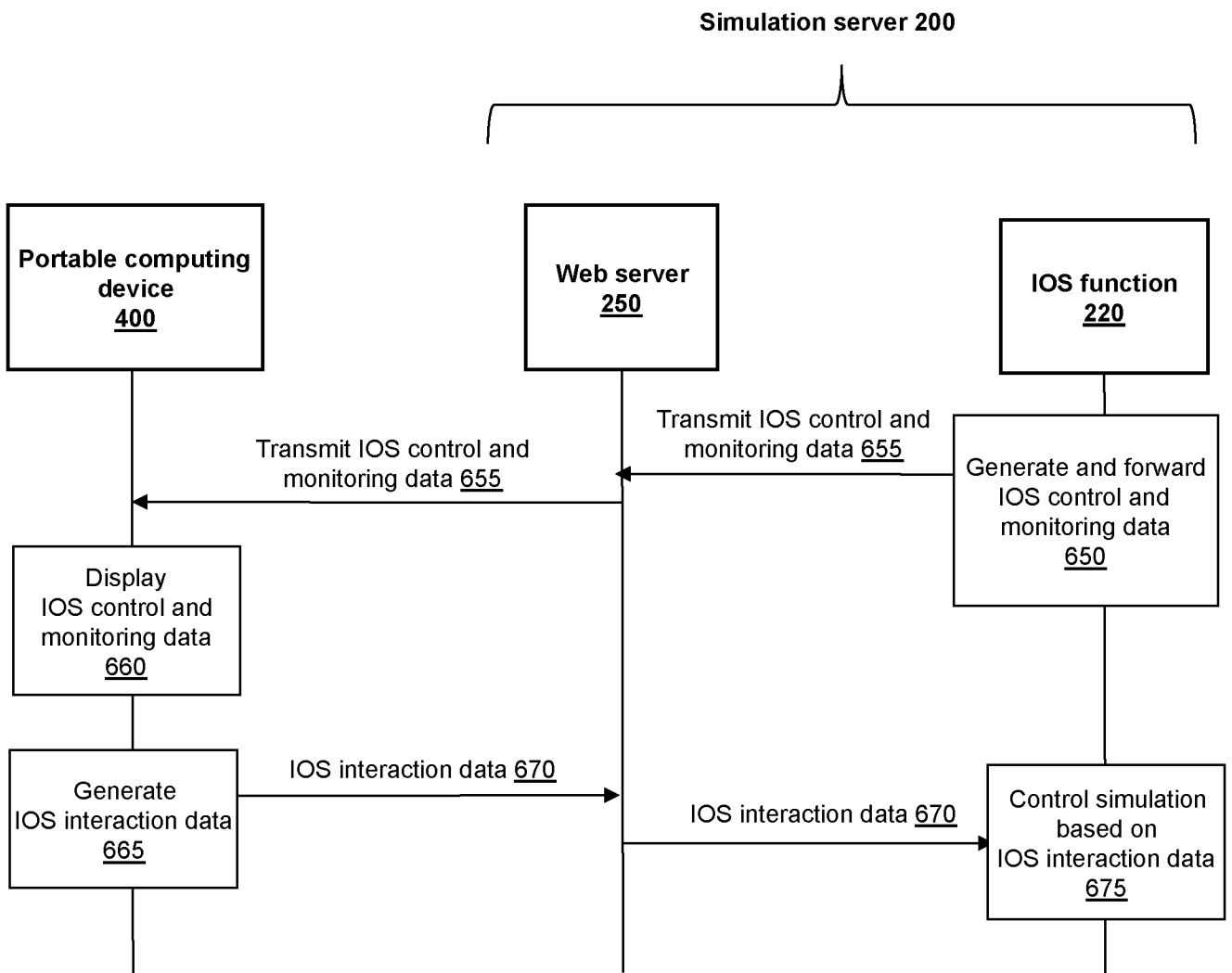


Figure 6C

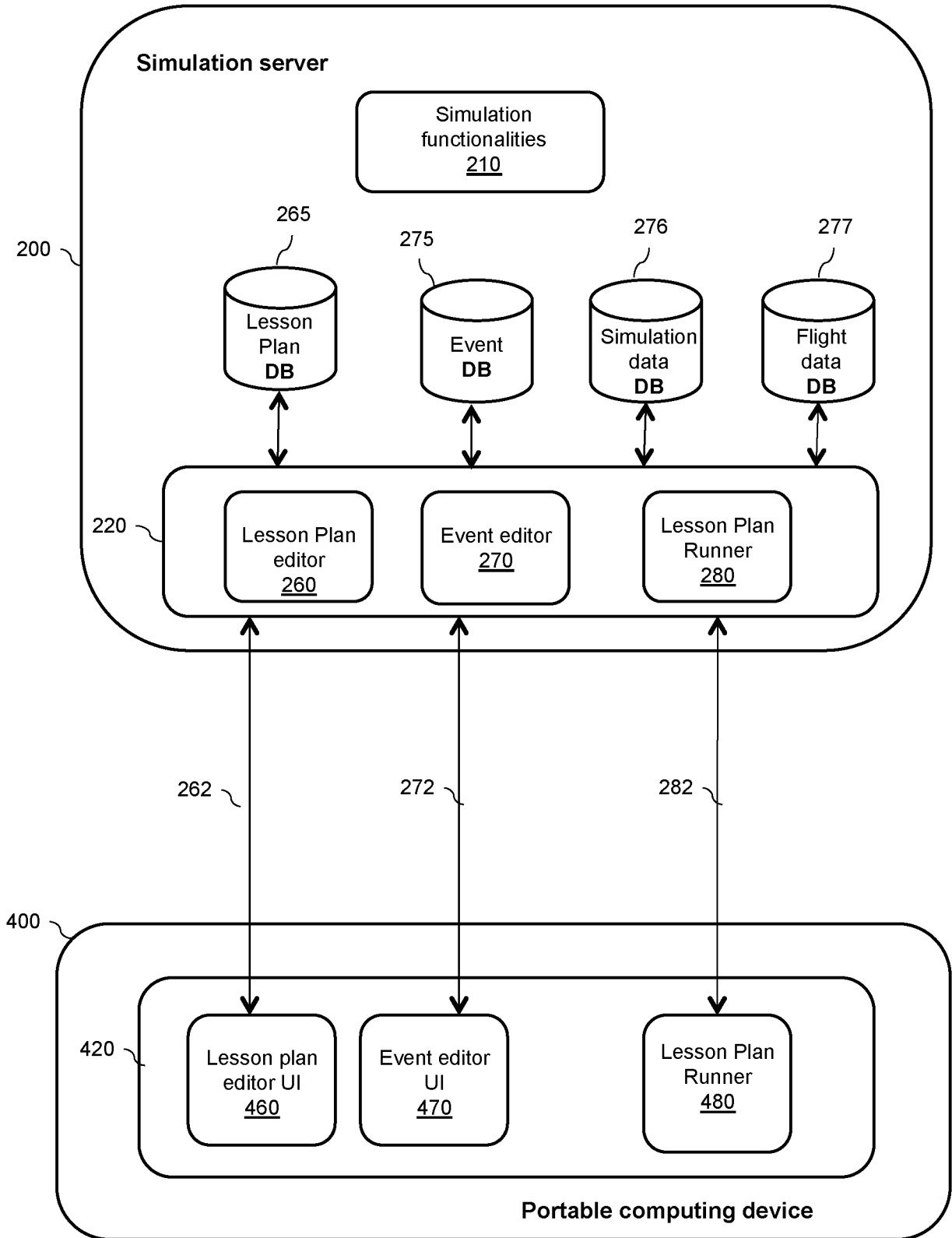


Figure 7

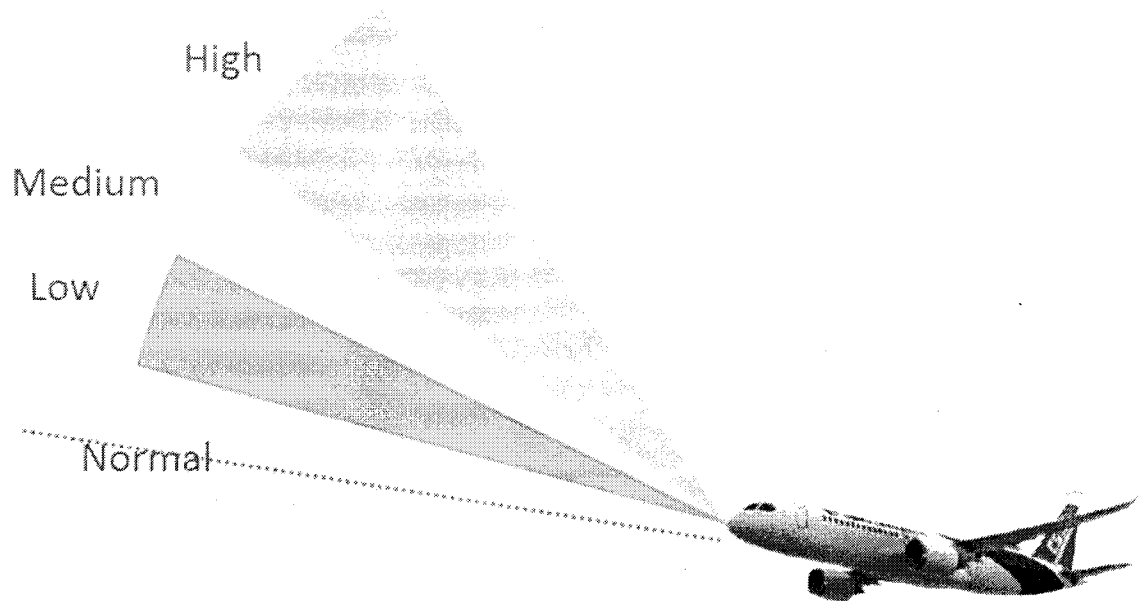


Figure 8

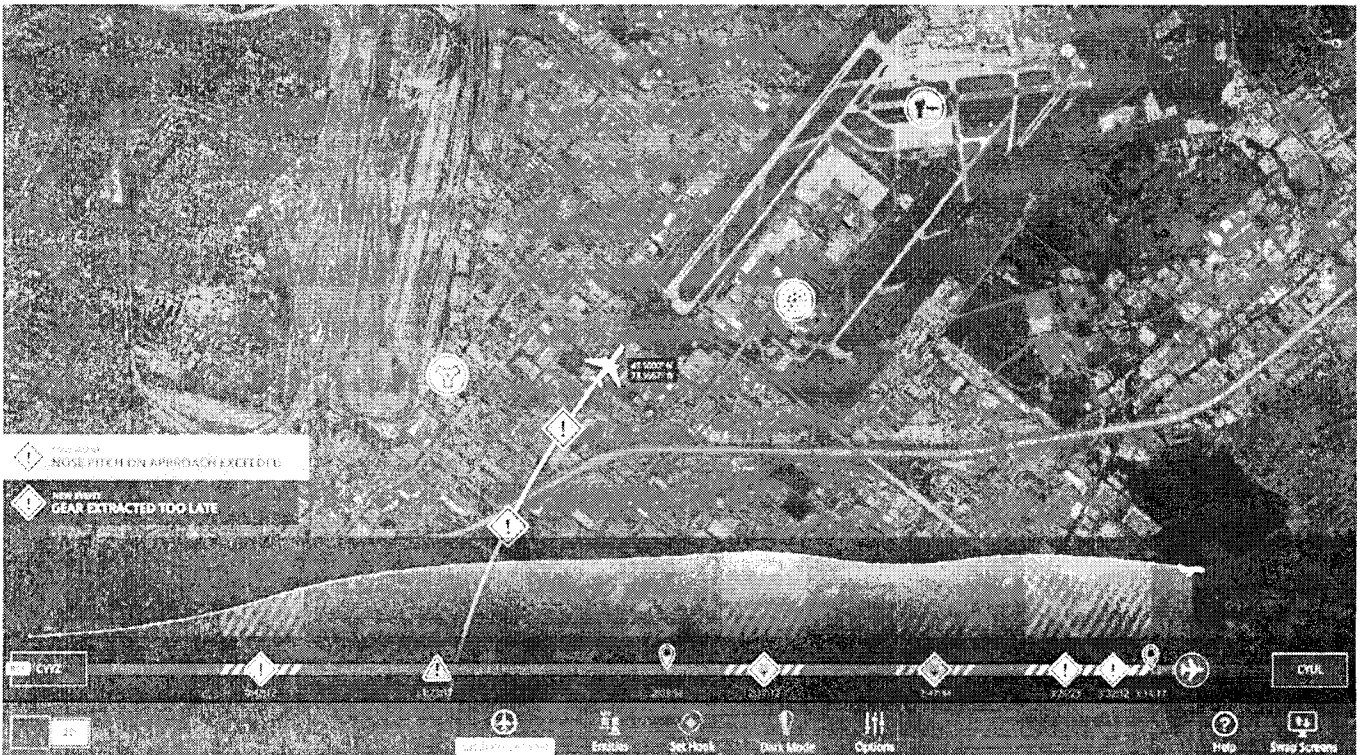


Figure 9

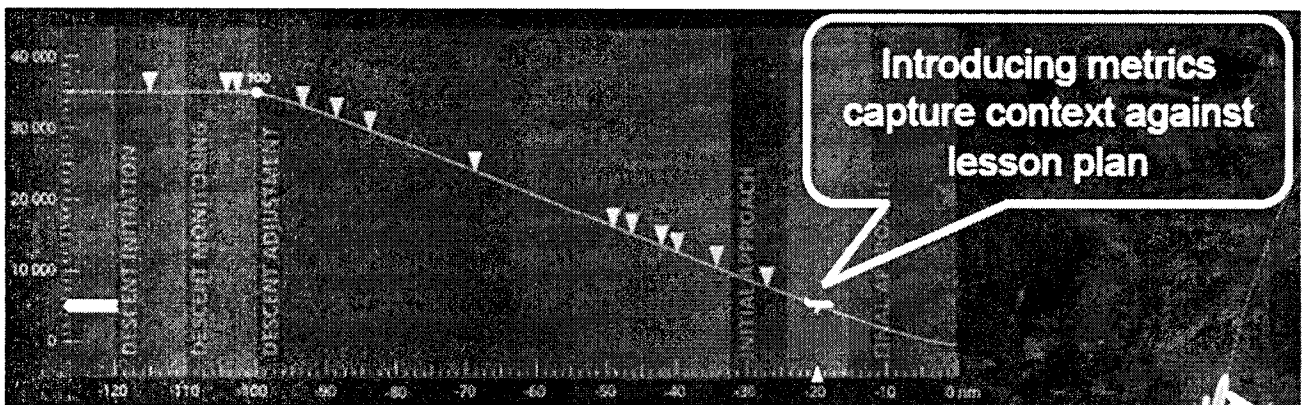


Figure 10

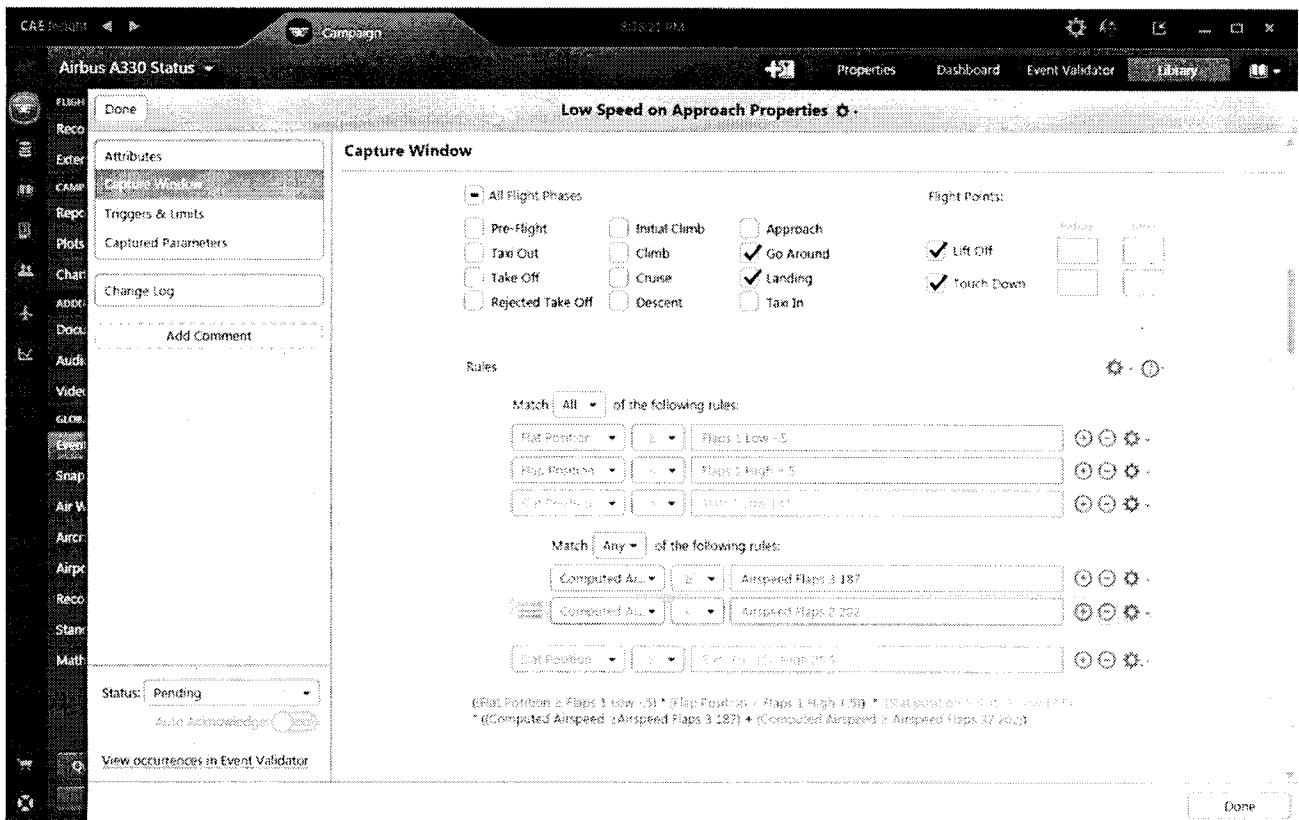


Figure 11

INTERNATIONAL SEARCH REPORT

International application No.

PCT/CA2017/050188

A. CLASSIFICATION OF SUBJECT MATTER
 IPC: *G09B 9/00* (2006.01), *G09B 9/08* (2006.01), *H04L 12/16* (2006.01), *G08G 5/00* (2006.01)

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
 IPC (2012.01): G09B, G08G, H04L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic database(s) consulted during the international search (name of database(s) and, where practicable, search terms used)
 Questel-Orbit, CIPO Intellect, Google Patents, IEEE XPLORE: flight AND simulat* AND server AND communication AND interface AND interact* AND data AND control and all such related terms.

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	WO 2015/143530 A1 MANSEAU et al. 01 October 2015 (01.10.2015) Entire document	1 to 17
Y	WO 2012/158432 A2 STACY et al. 22 November 2012 (22.11.2012) Entire document	1 to 17

Further documents are listed in the continuation of Box C.

See patent family annex.

* "A" document defining the general state of the art which is not considered to be of particular relevance "E" earlier application or patent but published on or after the international filing date "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) "O" document referring to an oral disclosure, use, exhibition or other means "P" document published prior to the international filing date but later than the priority date claimed	"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention "X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone "Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art "&" document member of the same patent family
---	--

Date of the actual completion of the international search
 19 May 2017 (19-05-2017)

Date of mailing of the international search report
 23 May 2017 (23-05-2017)

Name and mailing address of the ISA/CA
 Canadian Intellectual Property Office
 Place du Portage I, C114 - 1st Floor, Box PCT
 50 Victoria Street
 Gatineau, Quebec K1A 0C9
 Facsimile No.: 819-953-2476

Authorized officer
 Salvatore Ginese (819) 635-2023

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/CA2017/050188

Patent Document Cited in Search Report	Publication Date	Patent Family Member(s)	Publication Date
WO2015143530A1	01 October 2015 (01-10-2015)	None	
WO2012158432A2	22 November 2012 (22-11-2012)	WO2012158432A2 WO2012158432A9 WO2012158432A3 US2014109113A1	22 November 2012 (22-11-2012) 07 February 2013 (07-02-2013) 28 March 2013 (28-03-2013) 17 April 2014 (17-04-2014)