

CLAIMS

What is claimed is:

1. A method, comprising:
 - identifying a payment process corresponding to a payment service, the payment
 - 5 process including a logic flow of steps each corresponding to a functional operation;
 - compiling each of a plurality of programming language-based descriptions into a
 - distinct computer-executable code segment of a plurality of code segments, wherein
 - the plurality of programming language-based descriptions each describe a distinct
 - functional operation of a plurality of functional operations usable to implement the
 - 10 payment process and, wherein the plurality of code segments correspond to a plurality
 - of states of a finite state machine;
 - determining a transition relationship between states of the plurality of states
 - based, at least in part, on the logic flow of steps of the payment process;
 - generating a state transition table based, at least in part, on the plurality of states
 - 15 of the finite state machine and the transition relationship between the states;
 - storing the state transition table in a cache of a computer-based payment system
 - that implements the finite state machine; and
 - executing the payment process based, at least in part, on configuring the finite
 - state machine by retrieving the state transition table from the cache at a runtime of the
 - 20 computer-based payment system.
2. The method according to claim 1, wherein the plurality of states include at
- least one state that corresponds to two or more steps of the logic flow.
- 25 3. The method according to claim 1, further comprising:
 - writing the state transition table to a configuration file of the payment system for
 - the payment system to retrieve the state transition table from the configuration file of
 - the electronic payment system at a startup.
- 30 4. The method of claim 1, further comprising:
 - determining, through the finite state machine, a first state of the payment service;
 - determining that a first transition condition for triggering a transition from the
 - first state to a second state is met;

in response to the determining that the first transition condition is met, transitioning the payment service from the first state to the second state using the finite state machine; and

performing a functional operation corresponding to the second state.

5

5. The method according to claim 4, wherein the determining that the first transition condition is met includes querying the state transition table to retrieve the first transition condition.

10 6. The method according to claim 5, wherein the determining that the first transition condition is met includes:

retrieving, by the payment system, the state transition table from a configuration file of the electronic payment system at a startup.

15 7. A device, comprising one or more processors and a memory storing executable instructions of a program executable by the one or more processors to perform acts including:

identifying a payment process corresponding to a payment service, the payment process including a logic flow of steps each corresponding to a functional operation;

20 compiling each of a plurality of programming language-based descriptions into a distinct computer-executable code segment of a plurality of code segments, wherein the plurality of programming language-based descriptions each describe a distinct functional operation of a plurality of functional operations usable to implement the payment process and, wherein the plurality of code segments correspond to a plurality
25 of states of a finite state machine;

determining a transition relationship between states of the plurality of states based, at least in part, on the logic flow of steps of the payment process;

generating a state transition table based, at least in part, on the plurality of states of the finite state machine and the transition relationship between the states;

30 storing the state transition table in a cache of a computer-based payment system that implements the finite state machine; and

executing the payment process based, at least in part, on configuring the finite state machine by retrieving the state transition table from the cache at a runtime of the

computer-based payment system.

8. The device according to claim 7, wherein the plurality of states include at least one state that corresponds to two or more steps of the logic flow.

5

9. The device according to claim 7, wherein the executing the payment process is further based on conditions for transition between the states.

10. The device of claim 7, wherein the acts further include:
10 determining, through the finite state machine, a first state of the payment service;
determining that a first transition condition for triggering a transition from the first state to a second state is met;
in response to the determining that the first transition condition is met, transitioning the payment service from the first state to the second state using the
15 finite state machine; and
performing a functional operation corresponding to the second state.

11. A non-transitory computer-readable medium storing contents that, when executed by one or more processors, cause the one or more processors to perform
20 actions comprising:

identifying a payment process corresponding to a payment service, the payment process including a logic flow of steps each corresponding to a functional operation;
compiling each of a plurality of programming language-based descriptions into a distinct computer-executable code segment of a plurality of code segments, wherein
25 the plurality of programming language-based descriptions each describe a distinct functional operation of a plurality of functional operations usable to implement the payment process and, wherein the plurality of code segments correspond to a plurality of states of a finite state machine;
determining a transition relationship between states of the plurality of states
30 based, at least in part, on the logic flow of steps of the payment process;
generating a state transition table based, at least in part, on the plurality of states of the finite state machine and the transition relationship between the states;
storing the state transition table in a cache of a computer-based payment system

that implements the finite state machine; and

executing the payment process based, at least in part, on configuring the finite state machine by retrieving the state transition table from the cache at a runtime of the computer-based payment system.

5

12. The computer-readable medium according to claim 11, wherein the plurality of states include at least one state that corresponds to two or more steps of the logic flow.

10

13. The computer-readable medium according to claim 11, further comprising:
writing the state transition table to a configuration file of the payment system for the payment system to retrieve the state transition table from the configuration file of the electronic payment system at a startup.

15

14. The computer-readable medium of claim 11, wherein in the actions further comprise:

determining, through the finite state machine, a first state of the payment service;

determining that a first transition condition for triggering a transition from the first state to a second state is met;

20

in response to the determining that the first transition condition is met, transitioning the payment service from the first state to the second state using the finite state machine; and

performing a functional operation corresponding to the second state.

25

15. The computer-readable medium according to claim 14, wherein the determining that the first transition condition is met includes querying the state transition table to retrieve the first transition condition.

30

16. The computer-readable medium according to claim 15, wherein the determining that the first transition condition is met includes:

retrieving, by the payment system, the state transition table from a configuration file of the electronic payment system at a startup.