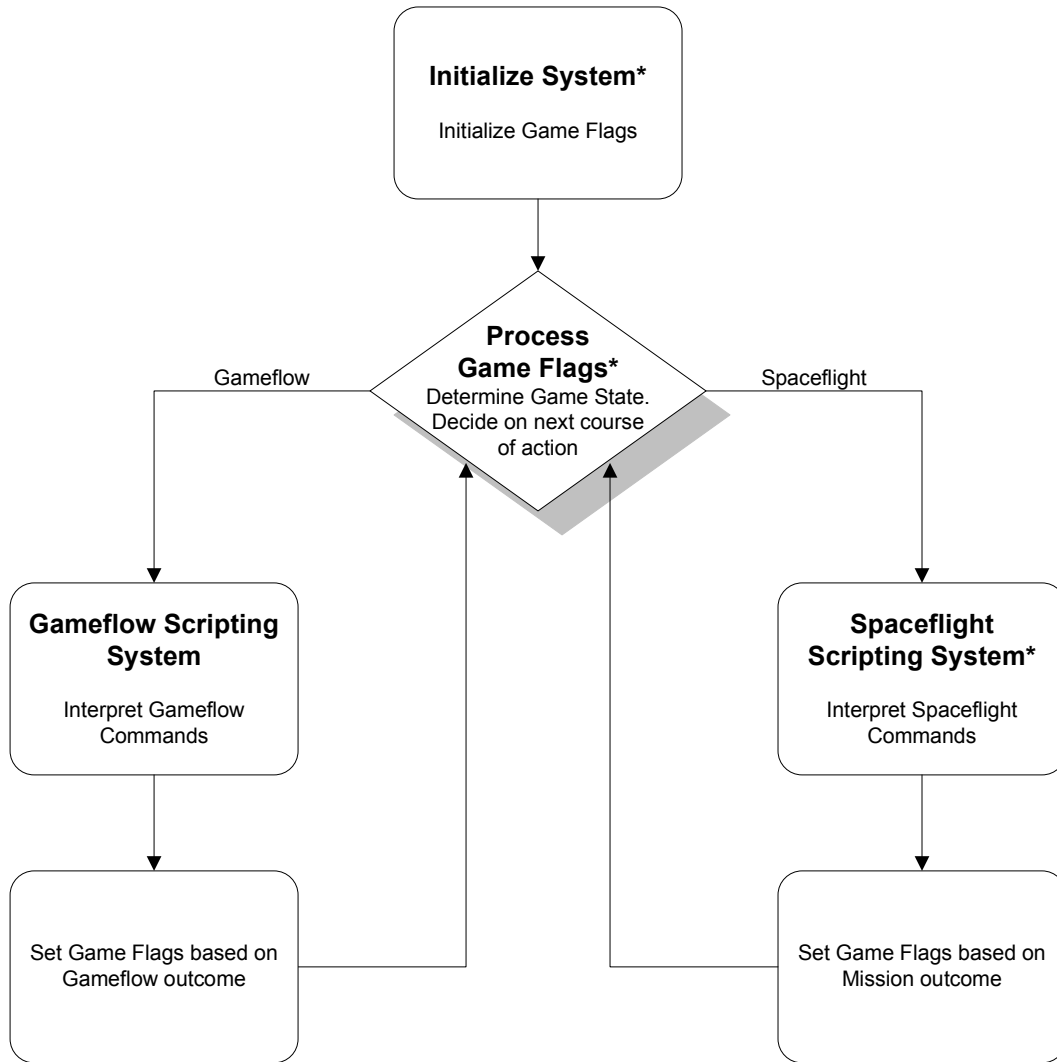


# Wing Commander V

## MCP Overview

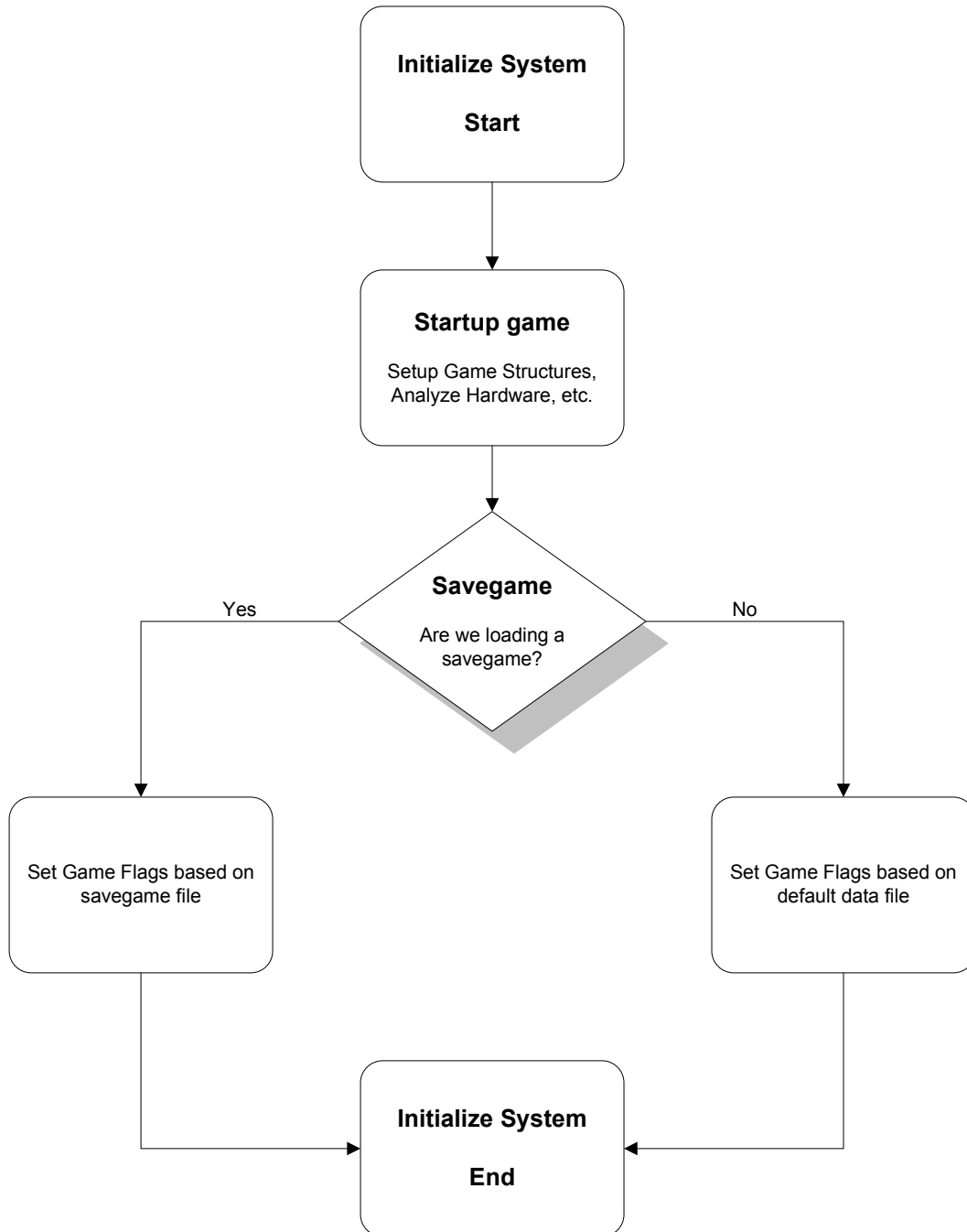


The MCP (Master Control Program) is designed to be the game moderator. The MCP is a data script that evaluates the global game flags to determine the state of the game, and decides the possible actions for the player, whether it be walking around a ship in gameflow, or flying a mission in spaceflight.

There is no implied linearity -- meaning the game does not have to alternate between gameflow and spaceflight. The MCP script can decide to play multiple missions in a row, or skip missions altogether.

# Wing Commander V

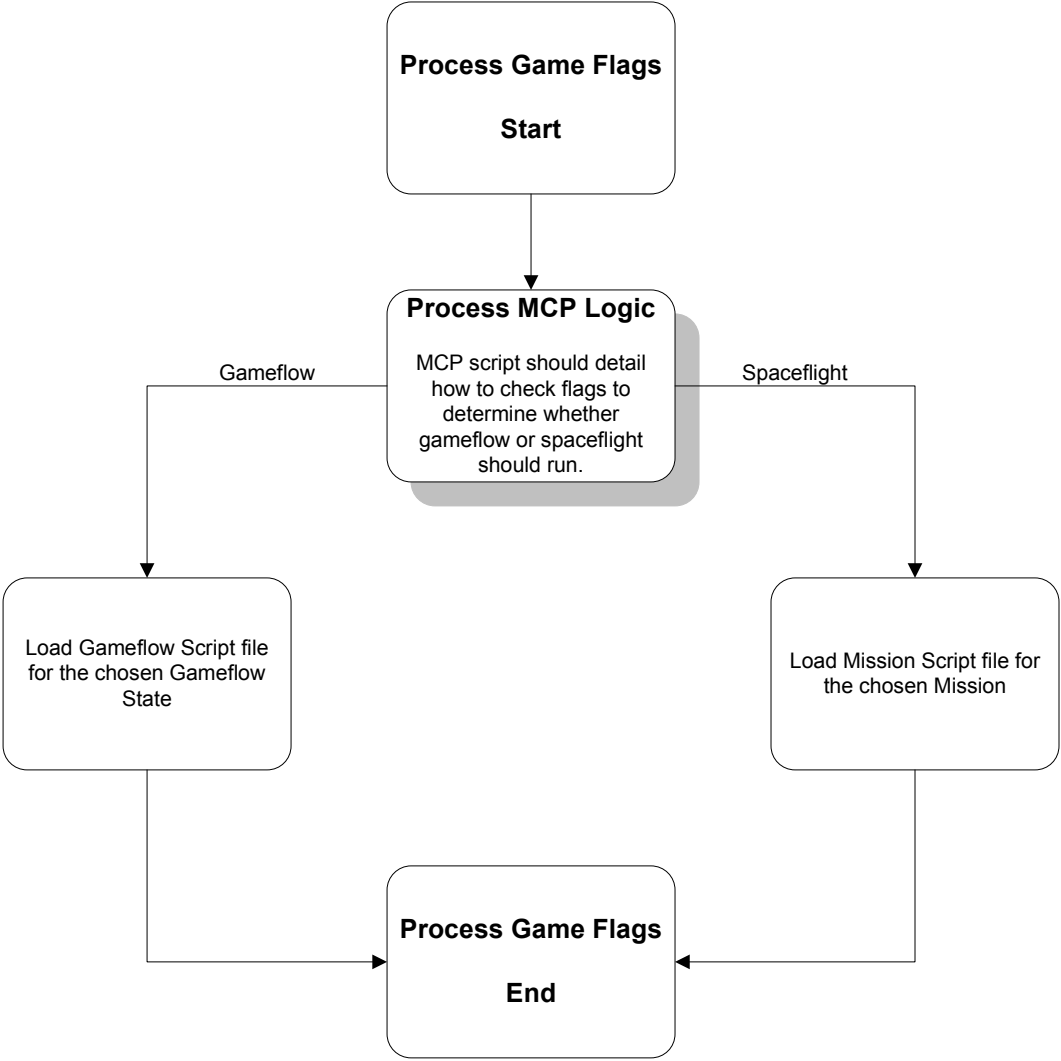
## MCP Initialize System



As far as the MCP is concerned, the main focus with *Initialize System* will be to know if the initial state for the MCP will be the default state, or a resumed state (most likely from a data file).

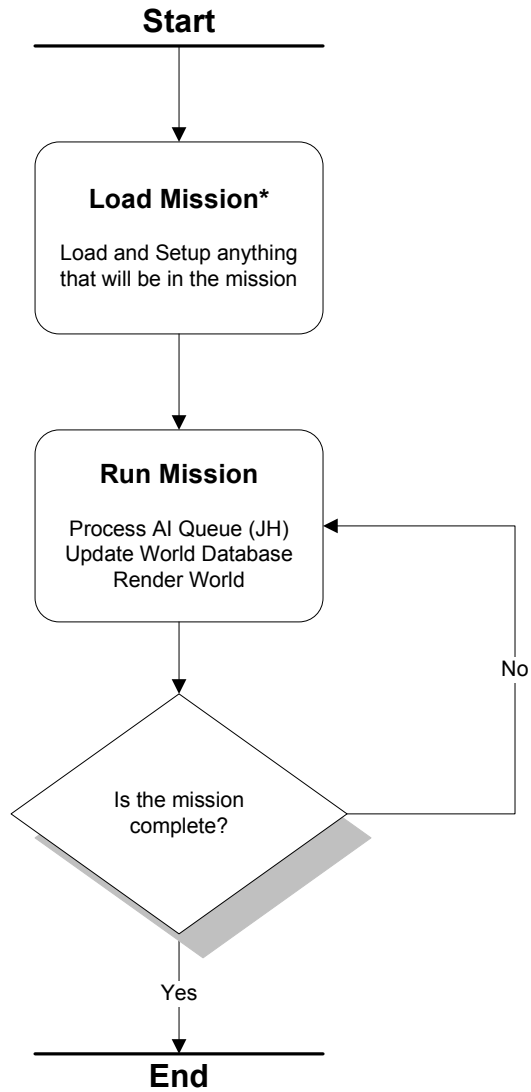
# Wing Commander V

## MCP Process Game Flags



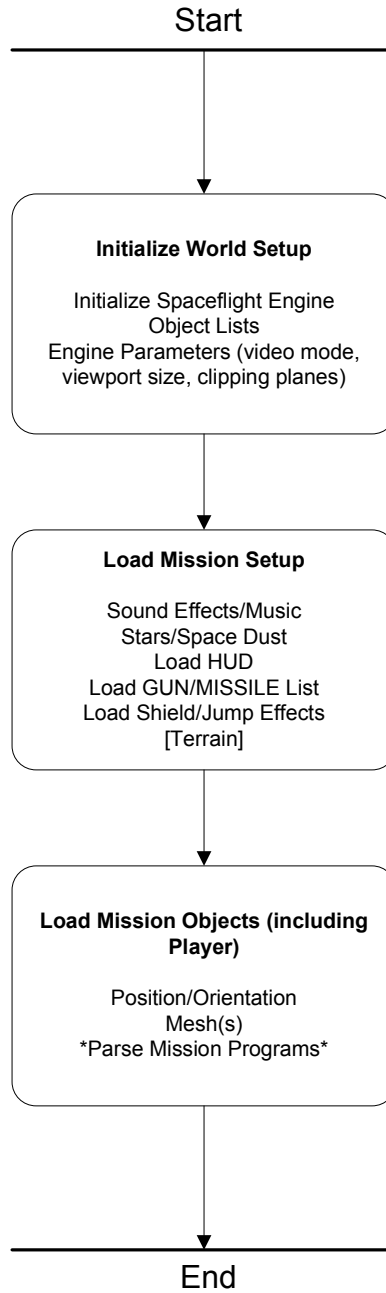
# Wing Commander V

## Spaceflight System Overview



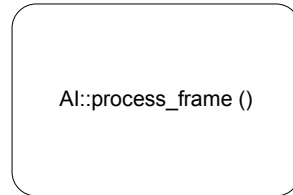
# Wing Commander V

## Load Mission

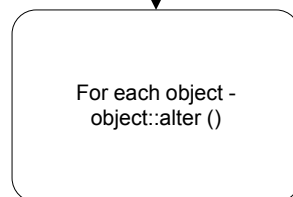


# Wing Commander V

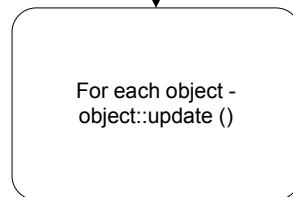
## Spaceflight System Main Loop (Run Mission)



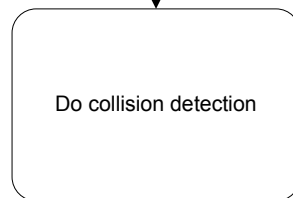
The AI scheduler determines what objects will receive an AI update. Only those objects receive new (or updated) AI commands.



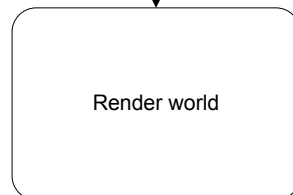
object::alter () sets up the internal flags/states of the objects in the world. These states will change when the object receives AI updates. Objects are not modified until object::update ().



All of the objects in the world will be updated (rotated/translated, etc.) based on internal states.



Collision detection will further update the positions/states of the objects in the world.



The world is rendered according to the active view.