This evolution introduces “the fog of war”—limited visibility of territories to those you control and adjacent ones. It also introduces a new unit type, the “spy,” which can grant you visibility to other territories you cannot see. The evolution 3 rules are:

1. Networked game play.
   (a) Multiple players (2–5 is required, more if you want) should be able to play, from different computers.
   (b) A client/server design must be employed.
   (c) The server is ultimately responsible for enforcing the rules of the game.
   (d) Clients may validate data in advance, but may not be trusted by the server.
   (e) A player disconnecting from the game may not unduly disrupt the gameplay of the remaining players. You may determine the specifics of what happens (forfeit, reconnection, etc), but the server may not crash, end the game, or make the game unplayable.
   (f) The server must support an arbitrary number of games at a time.
   (g) Each player should have a login and password.
   (h) Once a player has logged in, the server should send their client a list of the games they are currently involved in, as well as a list of games they might join (“the lobby”).
   (i) A player should be able to select a game (or create a new one), and play in it.
   (j) A player should be able to return from a game to the lobby.
   (k) The server must implement persistent state and resilience to crashes/power outages:
      i. If a crash (or power failure) occurs during game initialization, the server may discard the game or recover to a state where each player’s information state reflects some point in the initialization process from the client’s point of view (that is, a state that was in each client at some point prior to the crash).
      ii. If a crash occurs during the middle of a turn (while orders are uncommitted), it must recover to at least the state at the start of the turn.
      iii. (EC) For extra credit, the server may recover to a mid-turn state which is consistent with respect to the ordering of the orders that the each client sent the moves. That is, the list of orders must correspond to what the player’s client sent in some point of time during the turn.
      iv. If a crash occurs during the middle of a turn, the timeout must be reset to its maximum value (as if a new turn had just begun).
      v. If a crash occurs during the execution of the orders at the end of a turn, the server may either (your choice) recover to (1) the start of the turn, (2) the end of the turn—where all of the orders are committed, but none are executed yet, or (3) any state in the middle of the execution of the orders (as long as the state and remaining orders are consistent).
      vi. (EC) For extra credit, modify the previous requirement such that any outcomes which may have been reported to the clients prior to the crash are guaranteed not to change.

2. The game board shall be a map, divided into territories.
(a) During game play, the map shall be visibly displayed to each player.
(b) Each territory shall have a number of units in it.
(c) Each territory which is visible to a given player should graphically represent the pertinent state of that territory (who controls it, how many units are stationed there, resource production etc.)
(d) Each territory shall produce some number (0 or more) of resources of each type: food and technology.
(e) A territory shall be visible to a given player if one of three conditions is met: (1) the player controls that territory (2) the territory is adjacent to one controlled by the player (3) the player has a spy in that territory.
(f) A player shall receive no information about the status of a territory. If the user has previously seen a territory, the game may display the information from the last time it was visible in a way that indicates the information is stale.
(g) These visibility restrictions must be implemented in such a way that no players may cheat. Specifically, the server may not rely on the client to properly discard information that the player cannot see. The server must instead restrict the information it sends to each client based on what that player can see.

3. At the start of the game, an initialization phase shall take place where each player selects their starting territories and army placement:

(a) Territories shall be divided into initial starting groups (with the same number of territories in each group and the total of each type of resources produced across the territories in a group should be the same for all groups).
(b) Each player shall pick (or be assigned) one such group as her starting territories.
(c) Each player shall have the same number of initial units, which she may place in her territories as she wishes.
(d) Initial troop placement occurs simultaneously for all players, with no information conveyed between players about the other’s places until the process is complete. Put a different way, each player should be able to place their troops (and adjust that placement) until they are satisfied, this hit “Done” (or some similar button). All players can do this at the same time. Once all players hit “Done” their troop placements become visible to each other.
(e) The exact number of initial troops is up to you, or may be an option in setting up a new game. However, all players must receive the same number of troops.

4. Turn structure: A turn has four parts, which occur in the following order:

(a) Issue orders.
   i. There are three types of orders that a player may issue: move and attack (more below). The new move type is upgrade.
   ii. A player may issue any number of each type of these orders in a turn.
   iii. A player may revised/edit his orders until he commits them (by pressing “Done”, or similar button).
iv. No player may see the orders of any other player until all players’ orders are committed.

v. A player is free to revise (cancel, change, etc) his orders until they are committed or the turn times out.

vi. The server must ensure that all orders are legal.

vii. Once all players commit their orders, these orders are executed.

viii. Uncommitted orders must be sent to the server and stored on it.

ix. The game shall have a commit timeout which is configured when the new game is created. If this timeout expires before all players commit their orders, then all orders already sent to the server are automatically committed immediately.

x. This timeout restarts at the beginning of each turn.

xi. If there is a race with a client committing a order for turn N as the timeout expires, the server must ensure that the order is not accidently recorded as an order for turn N+1. That is, it must either include it in turn N if it is received before the timeout commit, or is discarded if received after.

(b) Execute orders: perform their effects

i. Move orders move troops from one territory to another territory controlled by the same player.

ii. A move order must specify the number of troops to move, the source territory, and the destination territory.

iii. Troops moving with a move order must have a path formed by adjacent territories controlled by their player from the source to the destination.

iv. Move orders effectively occur before attack orders.

v. Troops moving out of a territory do not participate in its defense.

vi. Troops moving in to a territory do participate in its defense.

vii. An attack order results in troops attacking a territory controlled by a different player.

viii. An attack order must specify the number of each type of troop to attack, the source territory, and the destination territory.

ix. Troops may only attack directly adjacent territories.

x. Attack orders effectively result in all attackers leaving their home territories simultaneously, then arriving at their attack target simultaneously.

xi. A successful attack (see “Combat resolution” below) results in the attacker taking ownership of the territory.

xii. All moves must follow the rules of common sense and preclude cheating; orders may not create new troops nor allow a troop to be in two places at once (attacking two territories).

xiii. Upgrade orders happen before any other order types.

xiv. A unit may be upgraded and move, or may be upgraded and used to attack/defend in the same turn.

xv. A spy may move into an enemy-held territory, however, a spy must only move one territory at a time while in enemy territory.
(c) Receive new units: At the end of each turn, one new infantry unit shall appear in each territory as long as the player who controls that territory will have enough food that no units will starve to death.

(d) Receive resources, feed troops, catch spies:

i. There are two types of resources: food and technology.

ii. For territory held, the player receives the amount of food and technology which the territory produces.

iii. Each unit (of any type) must be fed at the end of each turn, consuming 1 food.

iv. Whenever the player does not have enough food to feed all units (after receiving the food from territories held), units starve to death. The weakest (lowest attack modifier) units starve first. Ties between units of the same type are broken randomly by the server.

v. A spy in enemy territory has a chance of being caught (and killed) each turn he is in enemy territory. This chance chance starts at 1% on the first turn in enemy territory, and increases by 7% per turn. If the spy returns to friendly territory, this chance resets next time he ventures out to spy.

vi. Spies do not have an attack bonus, but are prioritized as if they had a +2 bonus when considering starvation ordering.

5. Combat resolution: The server should resolve each combat action, informing all players of the outcome.

(a) Combat between one attacker and one defender is an iterative process which ends when one side runs out of units in the fight:

i. The server rolls two 20-sided dice (one for the attacker, one for the defender). This roll shall be modified by adding the unit type bonus for the attacker’s unit and the defenders unit to their respective rolls.

ii. The side with the lower roll loses 1 unit (in a tie, the higher attack modifier wins. If there is still a tie, the defender wins and the attacker loses an unit).

iii. The order in which the units fight shall be a pairing which alternates between the defender’s strongest (highest modifier) unit paired with the attacker’s weakest (lowest modifier) unit, and the attacker’s strongest paired with the defenders weakest.

(b) If player A attacks territory X with units from multiple of her own territories, they count as a single combined force.

(c) If multiple players attack the same territories, the rounds (one opposed d20 roll) of attack are interleaved. For example, if A, B, and C attack territory X held by player D, then one round occurs between B and D, followed by one round between C and D, followed by one round between A and D. If the defender runs out of forces at any given time, whichever faction was involved in that round of combat against the defender becomes the new defender. Note that B/D, C/D, A/D all occur with defender high/attacker low, then an attacker high/defender low round occurs for B/D, C/D, A/D. The order of the attackers’ interleaving is randomized at the start of the fight, and whenever the defender changes.
6. Victory and defeat

(a) A player loses when he no longer controls any territories.

(b) When a player has lost, he may no longer make any moves, and the server should automatically consider his moves to be committed (as the empty set) at the start of each turn.

(c) A player who has lost may continue to watch the game if he desires, or may disconnect.

(d) A player wins when she controls all territories in the game.

(e) When a player has won, the server should announce this to all remaining clients, which should display this information. The game then ends.

(f) When a game ends, you may either have the server exit, or have it provide the option to start a new game.

7. Technology and upgrades

(a) Each player has a max technology level, which begins at “infantry.”

(b) A player may issue an upgrade order to research new weapons technology, increasing their maximum technology level. Doing so consumes their technology resource (and requires them to have sufficient technology resources to perform the action). The costs for each type are described in the following table:

<table>
<thead>
<tr>
<th>Level Num</th>
<th>Max Technology Level</th>
<th>Cost(Total)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Infantry</td>
<td>—</td>
</tr>
<tr>
<td>1</td>
<td>Automatic Weapons</td>
<td>20 (20)</td>
</tr>
<tr>
<td>2</td>
<td>Rocket Launchers</td>
<td>50 (70)</td>
</tr>
<tr>
<td>3</td>
<td>Tanks</td>
<td>80 (150)</td>
</tr>
<tr>
<td>4</td>
<td>Improved Tanks</td>
<td>120 (270)</td>
</tr>
<tr>
<td>5</td>
<td>Fighter Planes</td>
<td>150 (420)</td>
</tr>
</tbody>
</table>

(c) Each player must upgrade in order (a player may not upgrade to fighter planes until she first upgrades to Improved Tanks). These must be serialized by at least one turn. That is, upgrading from Tanks to Improved Tanks must be committed and executed before the player can upgrade from Improved Tanks to Fighter Planes.

(d) A player with a max technology level may issue upgrade orders for specific units to upgrade their types’ cost. The following table shows the per-unit upgrade costs, as well as the combat modifier bonuses:
(e) A player may upgrade any number of units (so long as she has resources) in a turn, however, the upgrade type must have been committed and executed in a previous turn in order to be available in the current turn.

(f) A player may upgrade a unit directly to any type available to her in one turn. For example, a player with Improved Tanks available may upgrade a unit with Rocket Launchers directly to an Improved Tank unit in one turn by paying 44 technology (55 - 11).

(g) The main game map display shall show the current max technology level obtained by each player which has been observed by this player. That is, if player X has seen player Y with an Improved Tank unit at any point the game, then X has observed Y with Improved Tanks and should remember that upgrade.

(h) The main game map display shall show a player his own current resource totals, but not those of any other player.

(i) Any unit may be converted to a spy for a cost of 35 technology (treat as an upgrade order).

(j) A spy unit may be converted back to its original unit type for a cost of 5 technology, as long as the spy is in friendly territory at the time that conversion is requested (treat as an upgrade order).

(k) Conversion of units to spies is separate from the maximum technology level and may be done at any point in the game.

Groups of 2 may omit the “spy” unit (2:e, part 3; 4:bxv; 4:dx:v–vi; 7:i–k), or if they do implement them, they will be considered extra credit.