

Changelog

Trinov (2025-09-17)

AI

Fixed error when AIArmy object sets strategy for its armies. Before it only set strategy for the main army, now it sets the strategy for all armies controlled by the AIArmy.

AI controlled armies will now send their generals for training in the capital when possible. They will not send generals for training while they are at war with another faction. Also added a concurrent modification check in method isIdle to see if this will stop the map freeze that seems to happen.

New features

Translated the UI to Indonesian.

Balance updates

None

Fixes

Fixed error in the new AI routine for adjusting the production. I had forgot to add a check that it should not update production for neutrals as then the game crashes since they don't have workshops.

Fixed concurrent modification crash in new method hasIdleArmies. Also moved method isIdle from ArmyMethods to being inside the Army class.

Fixed yet another crash in method expand when the expand AIArmy object has no army assigned to it.

Fixed concurrent modification crash in method reinforceAssets.

Fixed crash when clicking to open the next combat report but there are no combat reports to show.

Fixed a typo in explanation for abbreviation SEW.

Fixed crash in improveRequest when there is no main AI for checking if new barracks/stable/warehouse needs to be constructed.

Fixed crash when no vibrator could be initiated for haptic effects. Fixed this in all classes that uses the vibrator.

Fixed crash in method getTreatyType when no diplomatic relation can be found for the target faction id. When this happens, it will respond with treaty type neutral.

Fixed crash in method getEmpire when world object is null in AppWorldMemory. When this happens, it will return null instead of crashing.

Fixed error when creating start positions but no map exists. It will now skip creating start positions. How this will affect the game creation will be fun to see...

Valid for Imperium: Aeternum – Trinov update

Fixed concurrent modification crash when setting status and updating map army movement events.

Fixed crash when opening empire activity and initializing the diplomacy view but the empire object in AppWorldMemory is null. How can this even happen? Anyway, it will no longer crash at exactly this place.

Added check in method getImage that the graphics has been initialised. Else null is returned.

Fixed error when starting game activity and no faction data exists (even though there are checks for this earlier). It will now show the default human icons for manpower and horses. Also added these checks when initializing army and settlement layouts.

Added fix to the idle armies' button for when it sometimes doesn't work. Now it centres on the current active army asset, if any.

Improved memory handling by recycling the map before updating it in method updateMap.

Fixed a map freeze when drawing the AIArmy type icon for AI controlled armies.

Fixed crash in updateResourceBar when empire object is null. Now the resource bar won't be updated as there is nothing to update it with.

Fixed crash in getRecruitLevelBitmap when empire object is null. Now defaults to std icon for recruit level.

Editor

None

Other

Major rework of the drawObject method in MapView to decrease the issues with freezing or crashing maps due to concurrent modification issues. Now dedicated local objects will copy and store the information needed to draw the specific object, decreasing the risk that another thread uses the asset objects at the same time as the draw thread.

Major rework of internal code to make it more stable. Removed AppWorldMemory from MainActivity and made it into a singleton for better stability in the code and less NDK errors. This was a lot of work...

Optimised updateArmyLayout in GameActivity to make it faster and not using main thread heavy operations as much (removed nearly all findViewById which takes much processing power).

Optimised updateResourceBar in GameActivity to make it faster and not using main thread heavy operations as much (removed nearly all findViewById which takes much processing power).

Added a method to report non-fatal map freezes to Crashlytics for the Warrior version.