



X3D Earth Requirements: Yumetech Proposals

Alan D. Hudson, Justin Couch, Stephen N. Matsuba

The authors propose that the following requirements be adopted for the X3D Earth initiative:

1. Provide a seamless space to face viewing experience of the Earth
 - Allow the user to go inside the Earth as well as view subsurface data like well and mine data
 - Local override of terrain mesh and imagery desired
 - Allows a proposed construction site to show changes
 - Bathymetry data should be available
2. Each participant should contribute computing resources
 - Bandwidth
 - p2p distribution of assets
 - processing
 - storage
3. Server Requirements
 - Provide a reference Server Architecture
 - Provide at least one Open Source Implementation
 - Multiple versions of X3D Earth should be possible
 - Chain of materials, but local servers can override a resource
 - Web3D provides a base level resource for terrain and imagery
 - Allows the distribution of private data, ie classified sources, commercial data warehouses
4. Client Requirements
 - Provide at least one open source implementation of an X3D earth client
 - Easy navigation
 - Planet centered navigation mode
 - Ground level navigation mode
 - Subsurface navigation mode
5. World State
 - Provide a mechanism for distributing world state
 - Example: Is a light turned on?
6. Chat System
 - Chat areas divided by some mechanism—perhaps regional divisions.

7. Display of Volume data registered to Terrain data
 - ISSUE: How to render geospatial correct, typically a cube but needs be a frustum?
 - NASA Use Case: Underground scans for possible moon base
 - Planet 9 Use Case: Animated weather/dispersion display from simulation in a city
8. Community-provided object authoring
 - Provide an easy art path for users to create content
 - Voting System to bring best assets up / avoid spam
 - Multiple overlays of data/objects subscribable by user
 - Enable data vendors for overlays like 3D Buildings, GIS information
9. Enable client implementors to differentiate themselves
 - By how well a layer is rendered?
 - For example, tree coverage(color, texture map, 3D objects)
 - Could have conformance issues.
10. Enable multiple planetary bodies to be viewed
 - Up to the Solar System scale
 - NASA Use Case: Be able to show a complete earth to mars mission
 - Show exploration missions on asteroids as well for mining
11. User selectable truth or synthetic view of data
 - Any derived visualizations should be controllable by the user so the raw data can be seen
12. Data Fusion
 - Easy to combine multiple data sources on top of the world
 - GeoRSS overlay is a good example
 - WMS/WFS/WCS Support
 - Positioning of GeoTIFF files
 - KML file display?
 - Can either directly support some of these or make sure the API's make it easy
 - Ability to import/merge DWG and IFC files
 - This might be a conversion to X3D or directly inlining
13. Ability to represent building internals
14. Semantically markup items to enable smarter agent behaviors
 - Example: denote what are doors/windows/stairs
 - We are not sure what ontologies to use
 - Should we create as a layer so the client can request different versions?