

Delivering Next Generation Simulation Today

Convergence Through Cloud Technologies





Agenda

- Emerging Requirements for Next-Gen, Cloud-Enabled Simulation
- Whole Earth Game Engines
- The Potential of Game Technology
- Challenges
- Summary & Questions



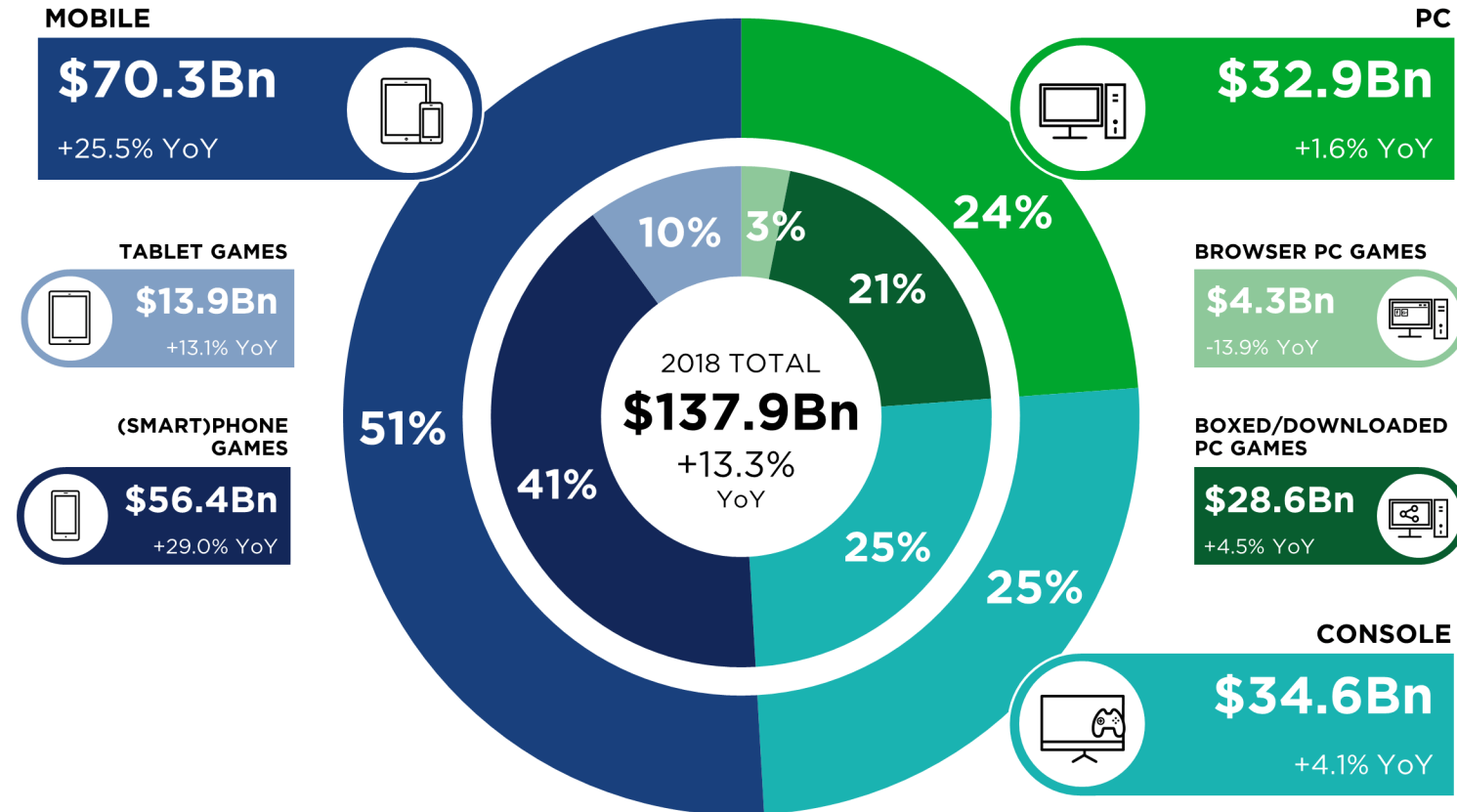


newzoo

2018 GLOBAL GAMES MARKET

PER DEVICE & SEGMENT WITH YEAR-ON-YEAR GROWTH RATES

©2018 Newzoo



Source: ©Newzoo | April 2018 Quarterly Update | Global Games Market Report

newzoo.com/globalgamesreport



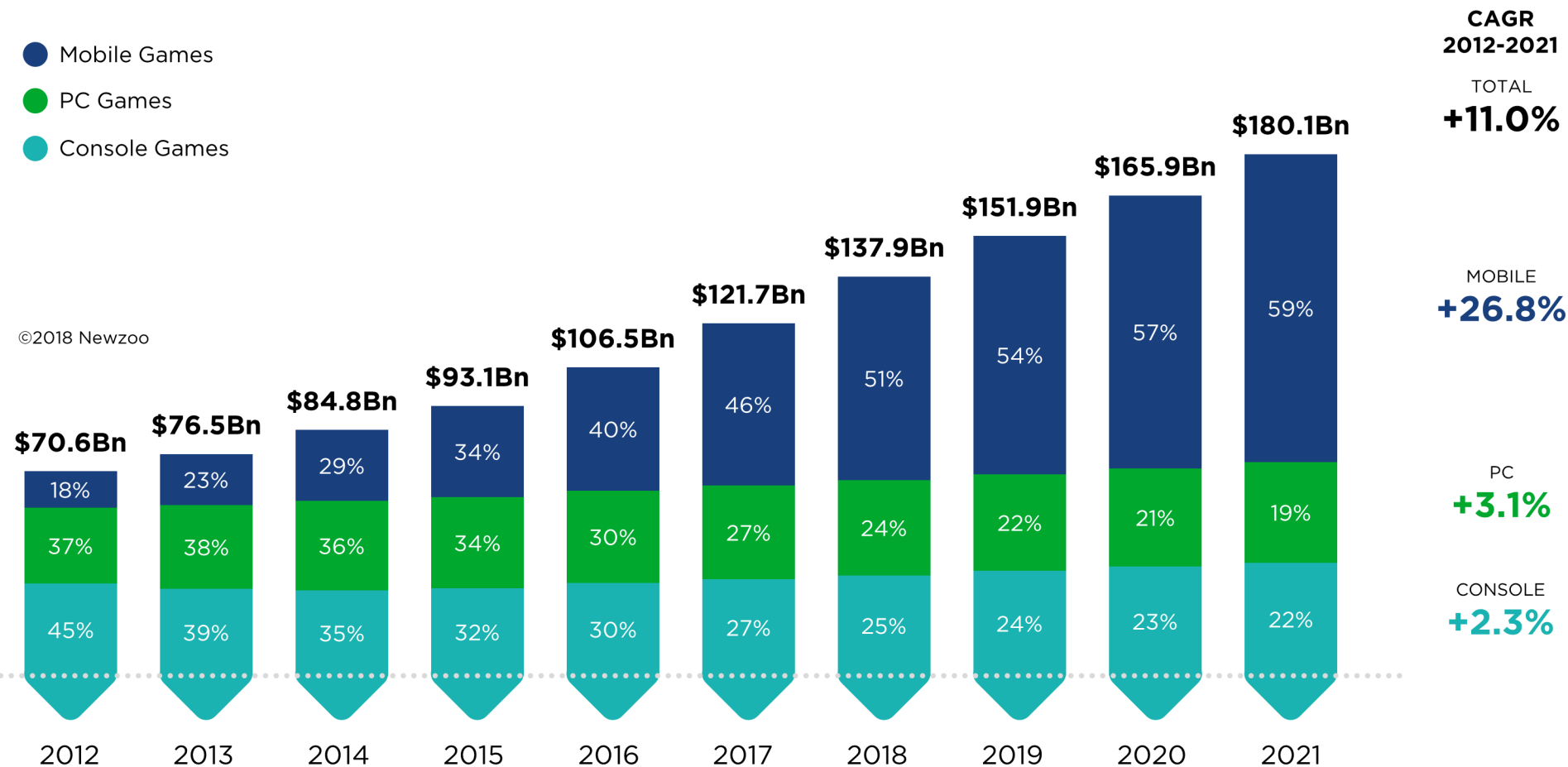
newzoo

2012-2021 GLOBAL GAMES MARKET

REVENUES PER SEGMENT 2012-2021 WITH COMPOUND ANNUAL GROWTH RATES

- Mobile Games
- PC Games
- Console Games

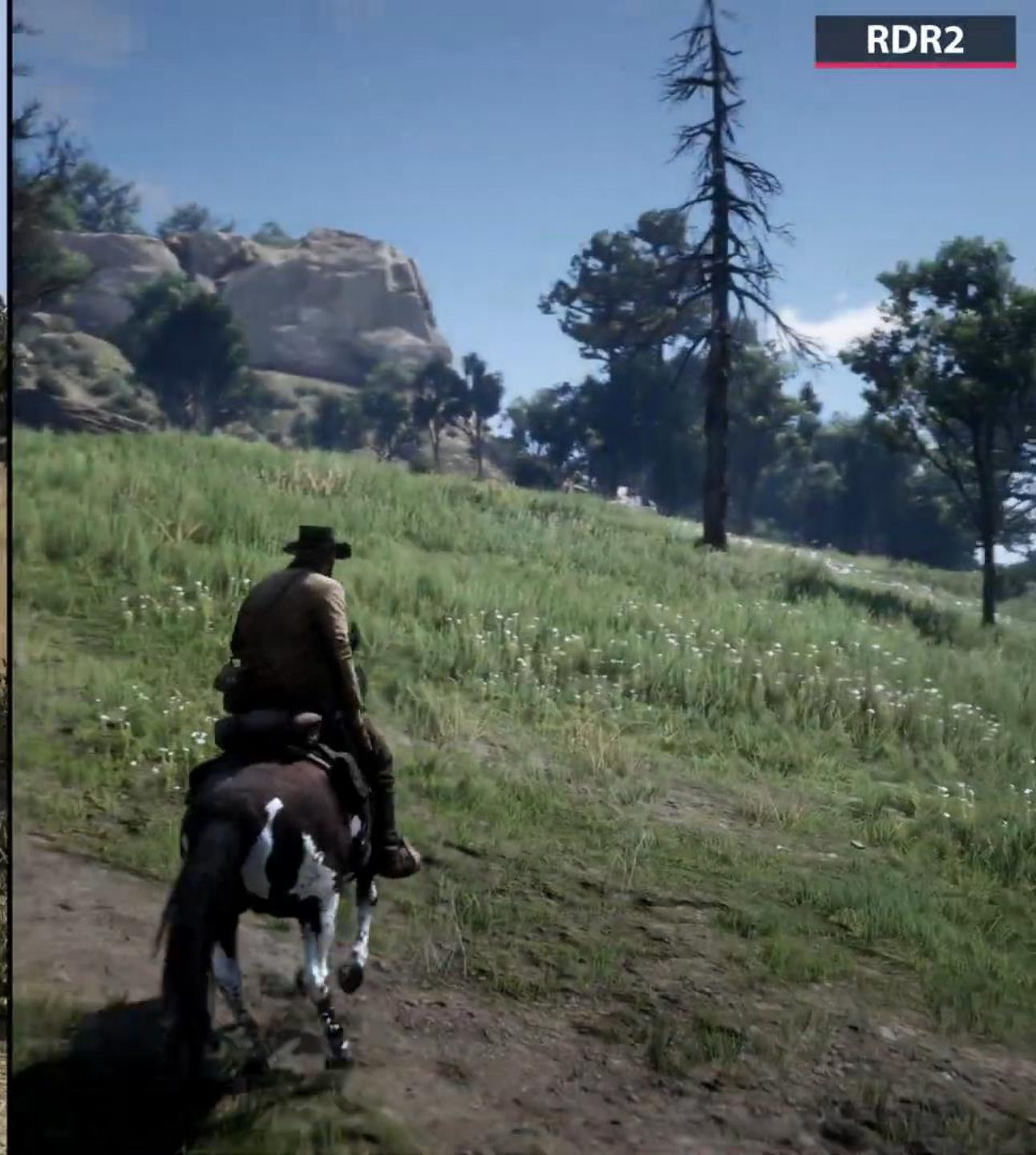
©2018 Newzoo



RDR1



RDR2







Why Games?

- The modular nature of game technology can help “future proof” future simulation deployments, **reducing the cost of implementing best-of-breed technology in the future.**
- Ultimately, component-based upgrades should be possible – even replacing the core “game engine”.



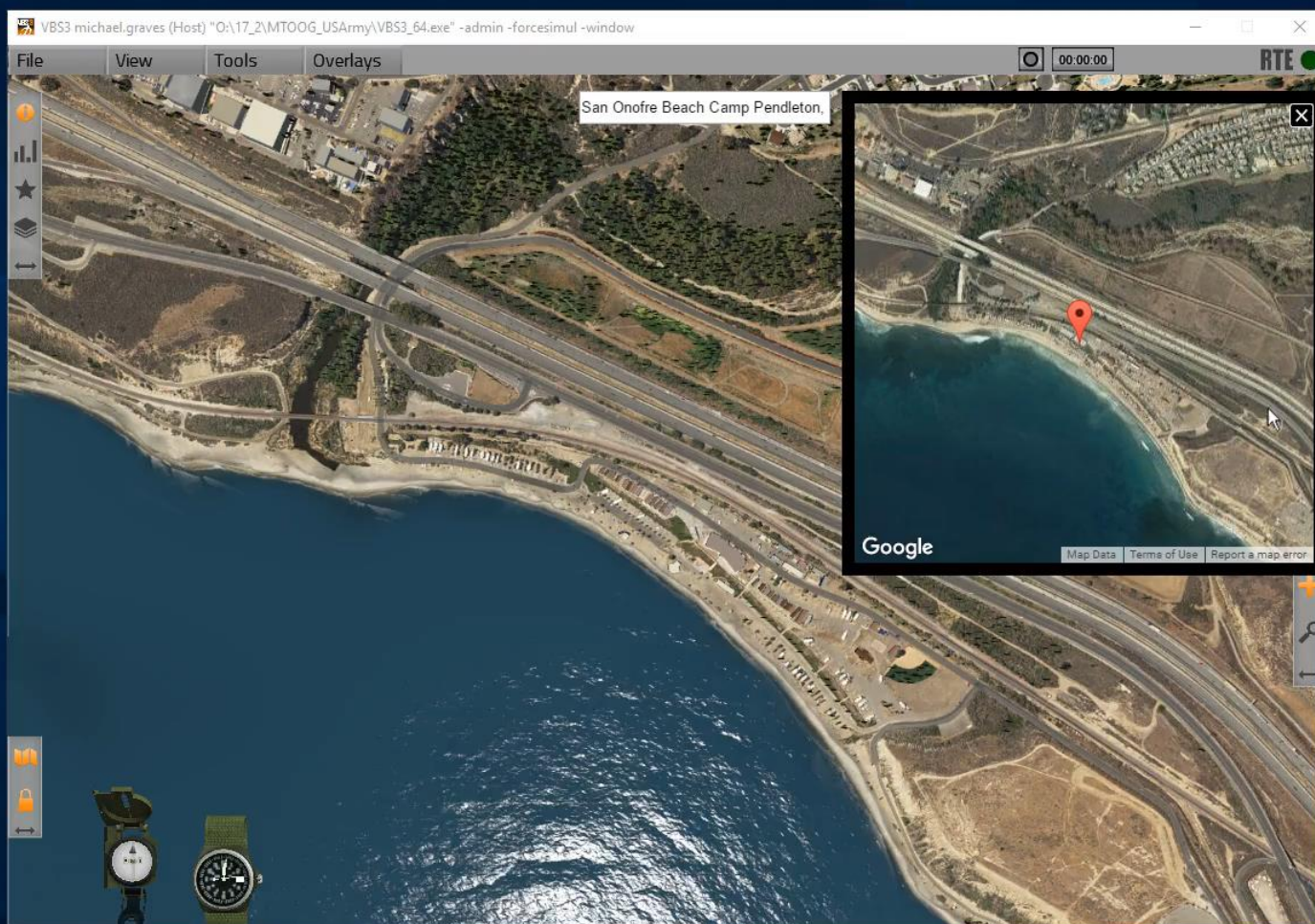
Example: Gears

- **Implementing modular game technology and broader (non-defense) industry standards will allow military organizations to more easily leverage the latest and greatest, and stimulate innovation.**
- Interoperability at individual application level can be achieved through **Gears**, a software development framework that defines a standard way for components to communicate through formal interfaces.
- Gears simply uses tools and approaches common to modern web development, where all software easily connect with each other by connecting DLLs (Dynamic Linked Libraries).
 - Modular software design has been around for many years. Dynamic Linked Libraries (.dll files) are a common part of any Windows-based software application.
 - A dynamic linked library is compiled source code that provides some function or capability, which connects to an application via an API.
 - *Gears is simply a way to more easily connect many dynamic linked libraries, to increase programming effectiveness and efficiency.*



www.gears.studio





GEARS STUDIO

Simplify and Standardize Your Software Development Process

[Download
Gears Studio Community Edition](#)

What is Gears?

Gears is a software development framework that defines a standard way for components to communicate through formal interfaces.

Gears uses a component-based architecture to promote rapid development by building applications from self-contained systems and having them communicate via formally defined interfaces. This allows functionality to be reused and avoids the complexity of tightly coupled systems.



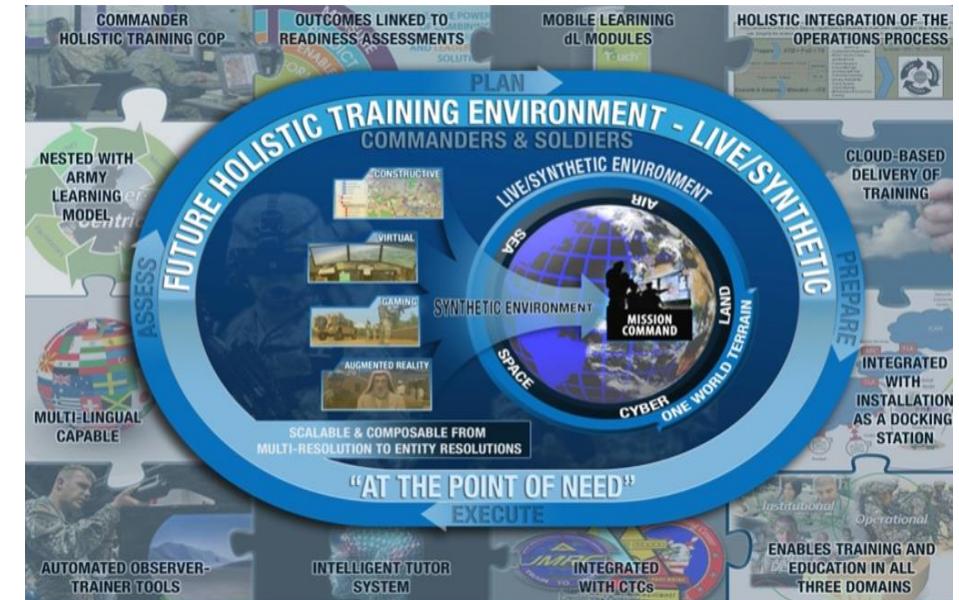
Gears Studio Overview

Gears Studio is a development environment to facilitate the development of products, components, and APIs within a framework intended to avoid common pitfalls and to automate common tasks encountered in software development.



Emerging Requirements

- Gaming technologies – especially those that leverage the Cloud – offer an unprecedented opportunity to *scale* simulation
- At least three premier militaries are considering Cloud deployment in future procurements:
 - United States Army (“Synthetic Training Environment” or “STE”)
 - United Kingdom Ministry of Defence (“Collective Training Transformation Programme”)
 - Australian Army (“Land Simulation Core 2.0”).
- The US Army STE program clearly articulates a need to “converge” virtual and constructive simulation, and deliver a holistic “One World Terrain” to the point of need, **via the Cloud**



Note: The appearance of U.S. Department of Defense (DoD) visual information does not imply or constitute DoD endorsement.



Emerging Requirements



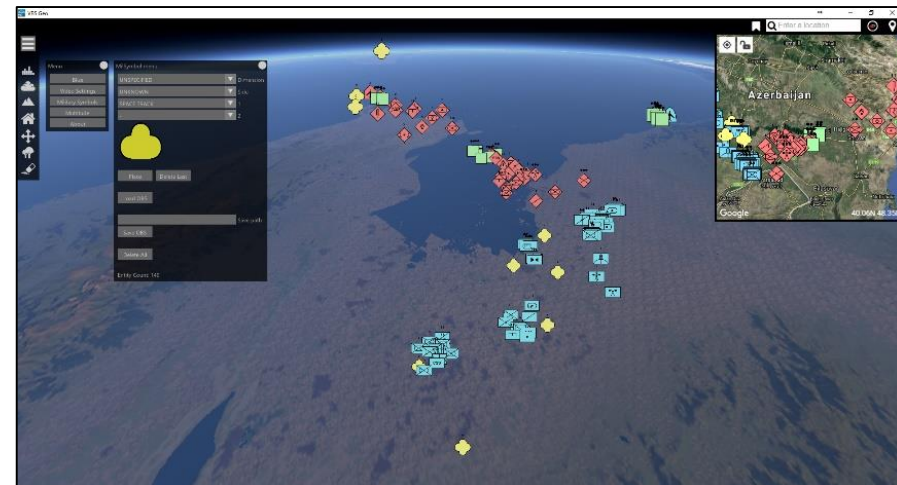
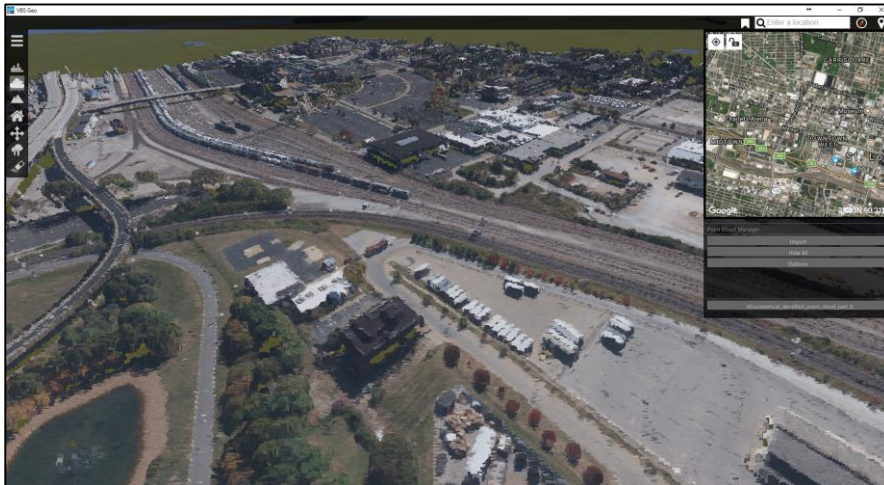
The following requirements are emerging among premier Western military forces.

Common, whole-Earth terrain representation <ul style="list-style-type: none">• Solves correlation issues between different simulation systems	Cloud-enabled <ul style="list-style-type: none">• Centralizing simulation capability• “One button” to start simulated training exercises
Connectivity to the point of need <ul style="list-style-type: none">• In-theatre forces training with those at home• Real-time terrain updates from in-theatre	Unprecedented scale <ul style="list-style-type: none">• Millions of simulated entities• “Massively multiplayer”
Persistent virtual world <ul style="list-style-type: none">• A persistent simulation of large regions of the Earth with cause and effect• Support for “non-kinetic” simulation	Convergence of virtual and constructive sim <ul style="list-style-type: none">• Soldiers and commanders training simultaneously• Seamless aggregation and disaggregation of entities depending on the training need
Aggregation of performance data <ul style="list-style-type: none">• All training systems should connect to a common LMS, facilitating Army-wide performance measurement and machine learning (ML)	Exploit Machine Learning <ul style="list-style-type: none">• ML informing changes to training scenarios based on human performance• ML for improved artificial intelligence



BISim and “Whole Earth Simulation” – A Brief History

- ☉ We began developing “whole earth rendering” tech in 2013 using IR&D exclusively – VBS Blue.
- ☉ We started doing funded R&D in 2017, prototyping “whole earth” concepts in VBS Blue. We developed multiple distinct demonstrations/technologies related to One World Terrain (OWT).
- ☉ VBS Geo, a prototyping application based on VBS Blue, was demonstrated by the U.S. Army at the STE Industry Day and I/ITSEC 2017.
- ☉ The following videos show our early work - PMESII layers in the prototype, and VR terrain editing.





Menu

Blue
Video Settings
Military Symbols
STWS
Multitude
DE Layers
Location Info ON
About

M/Symbol menu

UNSPECIFIED
FRIEND
GROUND TRACK
UNIT
COMBAT
INFANTRY
-

Dimension

Side

1

2

3

4

5



Place

Delete Last

Load OBS

Save path

Save OBS

Delete All

Icon Count: 1





From Space to Blades of Grass...

- Whole-earth engines like our own “VBS Blue” use cube-mapping techniques to render realistic scenes (with realistic view distances) at any altitude.
- Many modern computer games use similar techniques to deliver “whole planet” experiences...





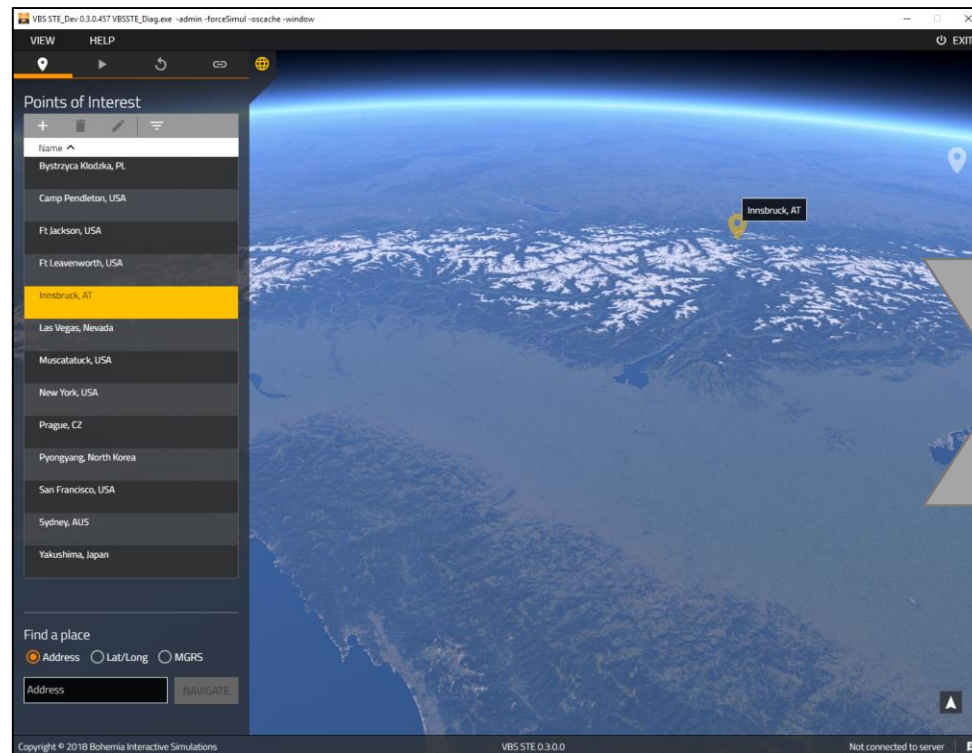


Shaping the Roadmap



Emerging requirements in programs such as STE are shaping the BISim technology roadmap...

- VBS Blue whole-earth game engine
- STE World Server for whole-earth paging terrain
- VBS Control AI framework
- Support for both Cloud **scalability** and Cloud **deployment**





Challenges

⌚ These new requirements are very challenging!

⌚ STE (for example) needs to be rolled out to a massive number of U.S. Army simulation systems.

- Likely to cost millions of \$ in integration costs for each system (e.g. CCTT, AVCATT, CFFT).
- Networks will need to be re-architected to integrate with OWT.
- Integration with live training / operations is also critical – further increasing bandwidth requirements

⌚ Traditional interoperability standards don't readily transfer to the cloud.

- DIS (and other multicast protocols) weren't designed for WAN operation.
- HLA requires expert knowledge and expensive technology (e.g. RTIs).

⌚ Cloud technologies are relatively new to the military.

- Security concerns
- Bandwidth concerns (e.g. 10mbps required for pixel-streamed graphics, *per channel*)
- Cost concerns (e.g. virtualizing a graphics card is still expensive!)

⌚ There is a desire to leverage game technology, but a lack of understanding how to employ it...

- Government desires ownership of source code, but still wants to leverage the best the game industry has to offer
- Rendering whole-earth terrain, representing whole-earth roads, rivers, bridges, buildings, etc., is not supported by many engines.



Summary

- “Game Technology” is vast and *could* offer higher fidelity or cost effectiveness compared to traditional approaches, depending on the training need!
 - Game “middleware” supports modularity
 - Cloud-based technology behind games like Worlds Adrift could accelerate STE
- Projects like STE are highly ambitious and remain a moving target. However the opportunities are many (for both Government *and* Industry), revolutionizing the way simulation is used in training.
- Modern industry open standards and technologies can address the challenges, and encourage innovation.
- Future approaches must be completely open – any simulation can interface through open APIs
 - Gears encourages software reuse.



Questions?

