



# Choosing Authoring Tools

*Webinar 3/25/09 12:00 – 1:00*

*ADL Co-Laboratory Hub*

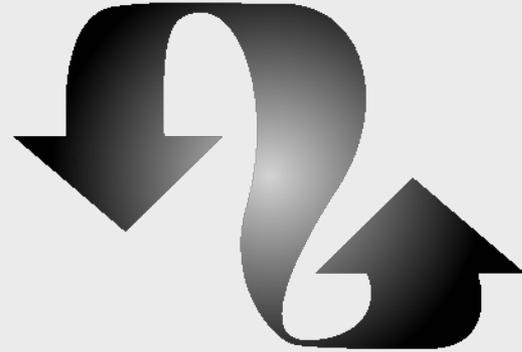
*Presented by: Peter Berking*

***Vendor citations or descriptions for illustrative purposes and not an endorsement by ADL.***

***All listings of vendors and products are in alphabetical order unless otherwise noted.***

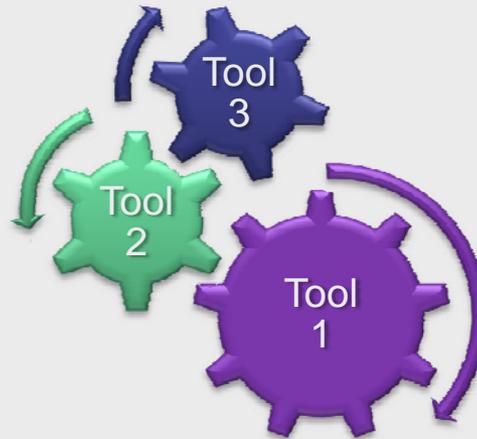
# Presentation Purpose & Scope

- Present considerations for choosing tools.
- Categorize tools and provide examples.
- Not intended as:
  - Comprehensive survey of available tools
  - Comparative rating or evaluation of products



# Presentation Purpose & Scope

- Title is “Choosing Authoring Tools” since usually one product will not suffice.



- ADL recommends that you use tools that produce SCORM-compliant e-learning.

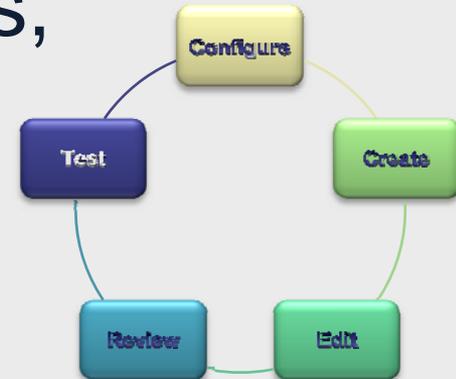
**SCORM**<sup>®</sup>  
Sharable Content Object Reference Model



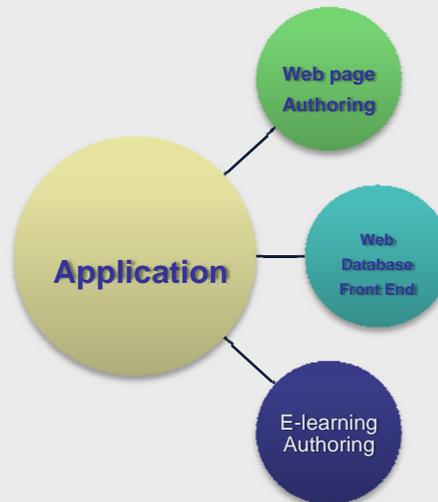
# Overview

# What Is an Authoring Tool?-1

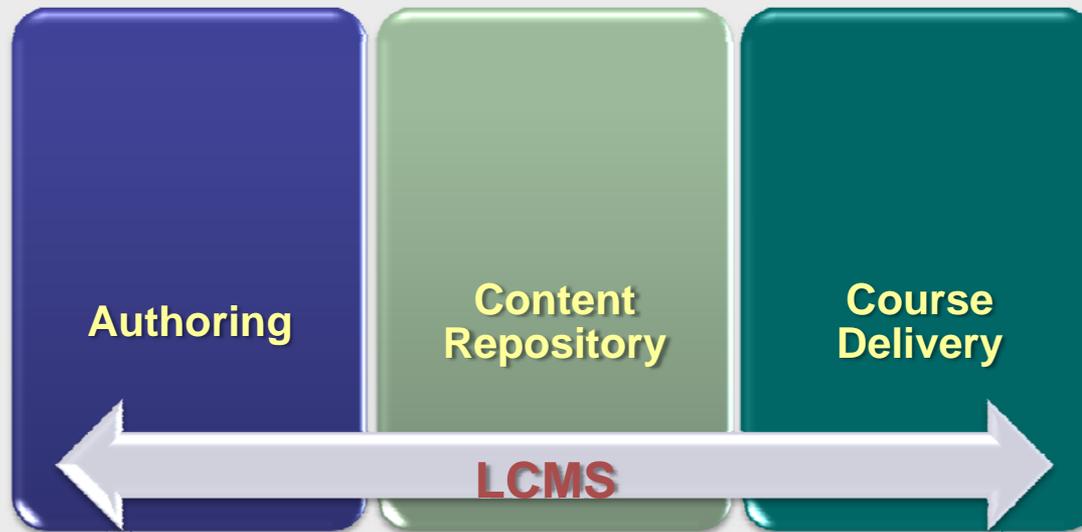
- Creates, edits, reviews, tests, and configures e-learning



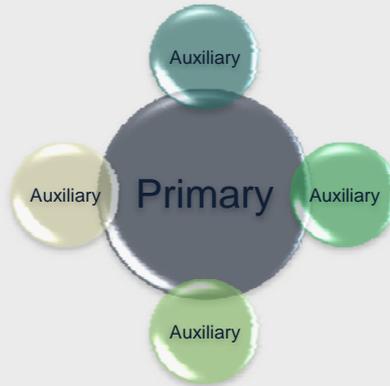
- Software may not be designed specifically for e-learning.



- Can be part of systems that perform broader e-learning functions (ex: LCMSs)



- Primary tools vs auxiliary tools

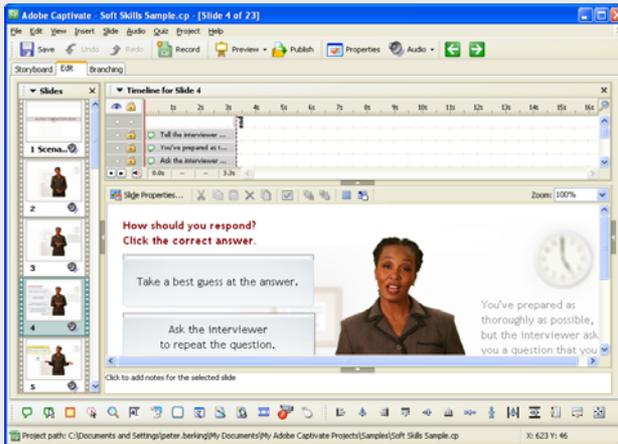


- Tools discussed here are primarily for WBT.



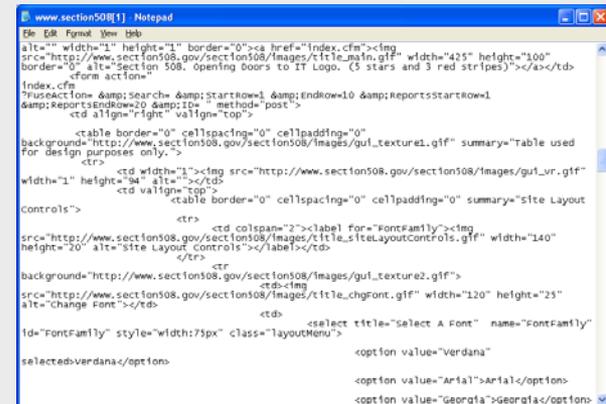
- Reduce technical overhead.

## WYSIWYG



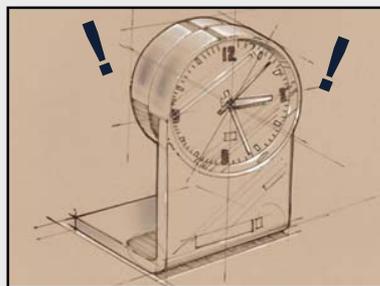
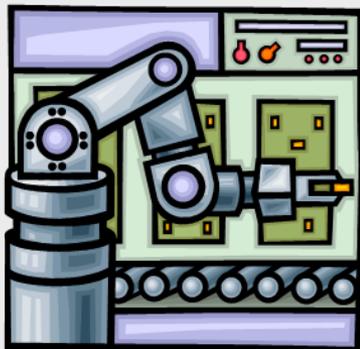
VS

programming or scripting language

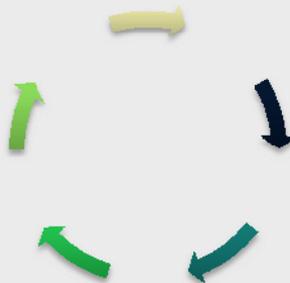


# Why Use Authoring Tools?-2

- Automate time-consuming tasks.



- Enforce standards and workflows.



# Why Is Choice So Important?

- Defines capabilities of your organization
- May or may not support your:
  - Types of e-learning products
  - Delivery platforms
  - File formats
  - Standards
  - Workflows
- Maximize durability.



## ■ Advantages:

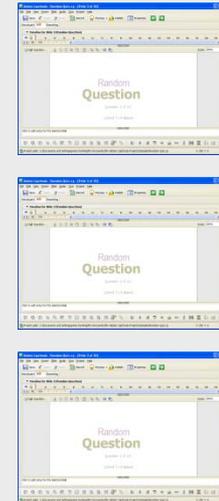
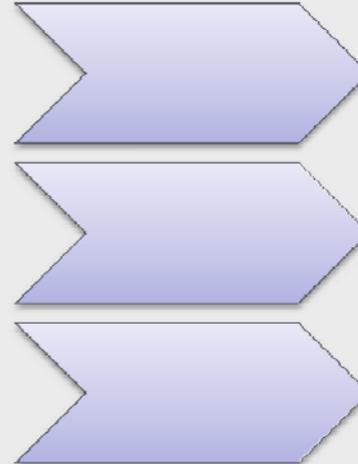
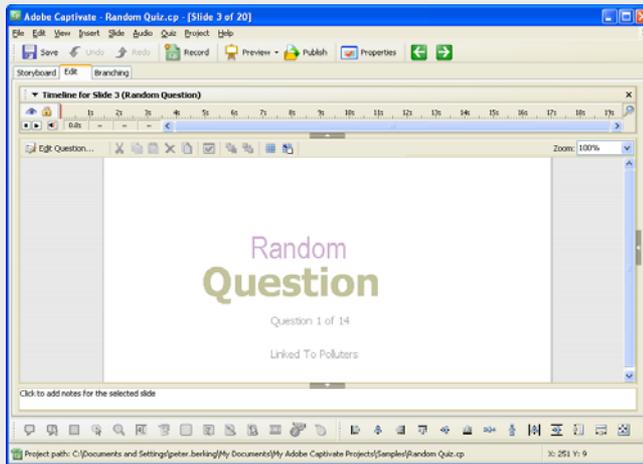
- Reduces costs through enterprise licenses



- Promotes economies of scale in training, help desk support, configuration management, etc.



- Advantages:
  - Enforces uniform standards through dissemination of application source file templates



- Disadvantages:

- Can unduly constrict e-learning appearance, functionality, or type of training



- Alternatives:

- Mandate standard tool for each type of training, format, etc.
- Implement style guides, templates, skins

- MIL HDBK 29612 provides guidelines that may be useful for tool standard specs.





# Process for Choosing Tools

## 1. Determine requirements for:

- ✔ Type(s) of training
- ✔ Media
- ✔ Level of interactivity
- ✔ Skill sets of authors
- ✔ Need for non-technical staff to edit content
- ✔ Output file format
- ✔ Standards compliance
- ✔ Kinds and levels of support and training required by the tool
- ✔ Collaborative authoring
- ✔ Number, roles, and distribution of potential tool users
- ✔ Bandwidth and other IT constraints and opportunities
- ✔ Budget for purchasing tool(s) and associated support/training contracts

2. Determine categories of tools you will need.



3. Identify specific tools for the identified key categories.



4. Develop matrix that assesses tools identified in step 3 against requirements developed in step 1.

*Tool Requirements Matrix*

	Media supported	Interactivity level	Authoring skill set	Editing by non-technical staff	Output file formats	Standards compliance	Support and training req'd	Collaborative authoring	Number, roles, and distribution of users	IT considerations	Budget
<b>Tool name</b>											

4. Filter and eliminate candidates.

6. Compile features list for all of the remaining candidate tools.
7. Develop and complete a matrix that weights features and identifies tools that have them.

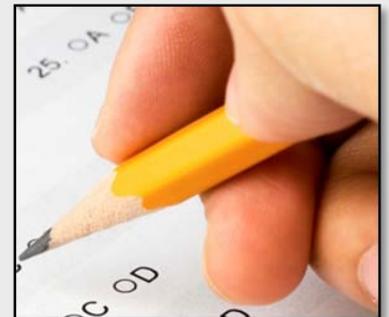
***Tool Features Rating Matrix***

	Feature 1	Feature 2	Feature 3	Feature 4	Feature 5	Feature 6	Feature 7	Feature 8	Feature 9	Feature 10	Rollup score
Tool name	Weighting factor										
											0
											0
											0
											0
											0
											0



8. Contact tool vendors and ask for detailed information, if needed to complete matrix.

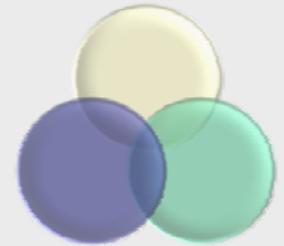
9. Make decision based on feature comparison, taking into account price, licensing, and other intangibles.



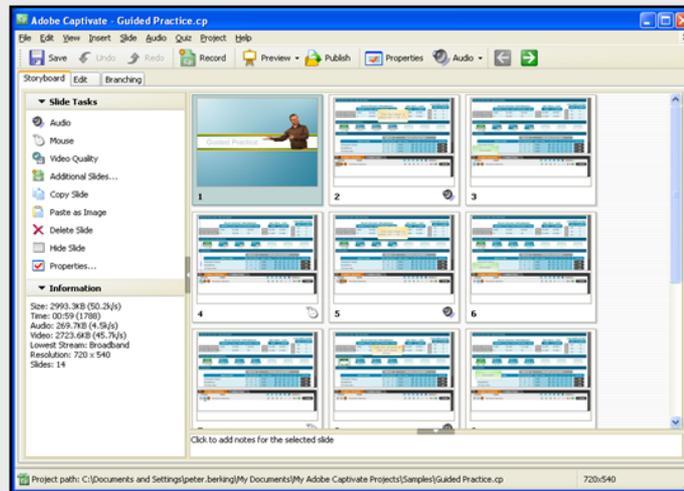


# Categories and Examples of Tools

- Purpose of categorization is to facilitate aligning your e-learning product requirements to tools.
- Categories not mutually exclusive.



- Self-contained authoring environments
  - Web development tools
  - Rapid Application Development (RAD) tools
  - E-learning development tools
    - Web-based e-learning development tools
    - Desktop-based e-learning development tools



- Self-contained authoring environments (continued)



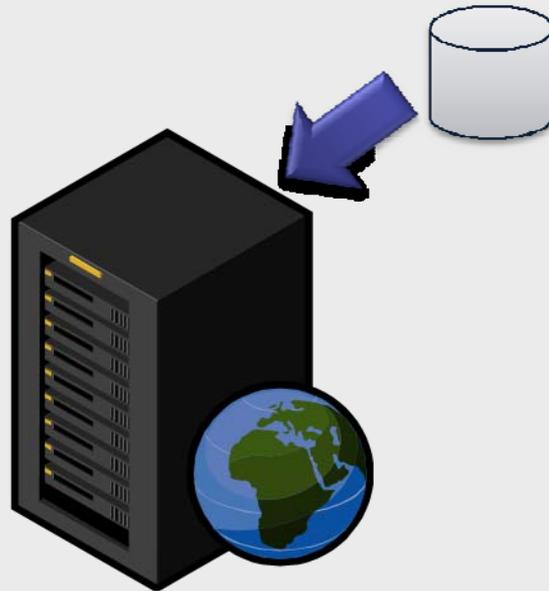
- Simulation development tools
  - System simulation development tools
  - 3D simulation development tools

- Game development environments



- Virtual world development environments

- Self-contained authoring environments (continued)
  - Database-delivered web application systems



- Learning content management systems (LCMSs)



- Virtual classroom systems



- External document converter/optimizer tools

- Web-based external document converter/optimizer tools



- Desktop based external document converter/optimizer tools

## ■ Auxiliary tools

- E-learning assemblers/packagers



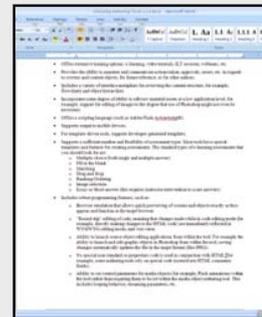
- Specific interaction object creation tools



- Auxiliary tools
  - Media asset production tools



- Word processors, page layout, and document format tools



## ■ Auxiliary tools (continued)

- Database applications



ID	Item	Category	Condition	Location	Contact	Acquired Date
1	56584626	(1) Category (1) Good	(1) Location			6/10
2	15672025	(1) Category (2) Good	(1) Location			6/10
3	27827822	(1) Category (1) Good	(1) Location			6/10
4	78278222	(1) Category (1) Good	(2) Location			6/10
5	20783783	(1) Category (2) Good	(2) Location			6/10
6	02783275	(1) Category (1) Good	(1) Location			6/10
7	78278222	(1) Category (1) Good	(1) Location			6/10
8	27827825	(1) Category (2) Good	(2) Location			6/10
9	78934025	(1) Category (1) Good	(1) Location			6/10
10	78979886	(1) Category (2) Good	(1) Location			6/10
11	33	(1) Category (1) Good	(1) Location			6/10
Total	113 Categories (1) Good					6/10

- Web-based collaboration and web 2.0 authoring tools



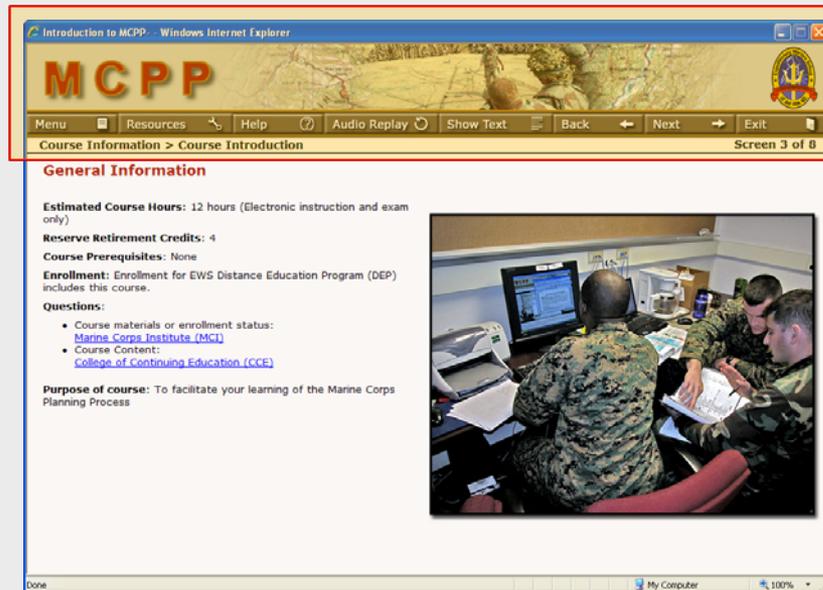
- Web page editors





# Major Features to Consider

- Definitions:
  - Skins are style sheets
  - Templates are starting point for building screen
- Can include a lot of functionality as well as appearance



- Advantages:
  - Saves time
  - Reduces requirement for technical expertise
  - Simplifies authoring process
  - Enables non-technical staff to author
  - Can enforce uniform standards
  - Skins enable local variations on parent content objects.
- Disadvantages:
  - Can restrict creativity and create “cookie cutter” look and feel

- Usually Word and PowerPoint files  
- External document converter/optimizer tools use them.   
- Steps:
  - Convert docs to web pages residing within an e-learning interface.
  - Build e-learning interactivity into these pages.
- Ensure that your legacy files match tools' supported input formats.

- Most important question: What output file format(s) does it produce?



- Establishing output format before choosing tools ensures that:
  - ✓ Files will work within your infrastructure.
  - ✓ You are not stuck with non-durable file format.
- LCMSs assemble and deliver files at runtime.



- Drivers for output format decision:
  - Type of training
  - Requirements of delivery system
  - Compatibility with network and firewalls
  - Compression and streaming features
  - Editability in other tools
  - Support for mobile learning era

- To support SCORM, the tool must:
  - Support object-based learning design.
  - Allow defining of SCOs at any level of organization.
  - Include a SCORM metadata editor.
  - Create SCORM course packages.
  - Allow definition of sequencing and navigation rules.



- For SCORM 2004 sequencing and navigation, use any of the following:
  - XML and Javascript
  - Reload Editor (<http://www.reload.co.uk>) (free)



- Features in tool (rarely available)
- Determine the level of compliance of your course delivery system.
- Test sample content on your target course delivery system to verify 100% compatibility.

- Run sample SCO produced by authoring tool through SCORM Test Suite.
- SCORM Adopter listing on the ADL public web site

<http://www.adlnet.gov/Technologies/scorm/Custom%20Pages/SCORM%20Adopters.aspx>



- Tool should facilitate generating compliant approaches/code/functions and disallow non-compliant ones.
- Some tools have built-in compliance checkers within the authoring tool.
  - Verify compliance by testing with screen reader.





# Criteria for Assessing Quality and Suitability of Tools

- Ease of learning and using, ideally with expertise tiers
- User interface customization
- Support for standards
- Support for media file formats
- Support for printing sets of course screens
- Integration of storyboarding process into tool
- Support for creating student surveys and certificates

- Support for wide variety of delivery architectures. Example: use of a content repository
- Support for creation of a desktop executable file
- Ease of installation and configuration
- Attainability of system requirements
- Management of production process with built-in workflows
- “Organization aware”

- Optimization for reusability in general
- Flexibility of licensing agreement
- Templates and/or skins features
- Ability to configure calls to external applications and code objects
- Support for wide variety of assessment types
- Ease of navigation and sequencing of content objects (ideally using SCORM 2004)

- Cost compared to competing authoring systems with the same or similar feature set
- Documentation and support
- Training options
- Ability to annotate and communicate actions taken, approvals, errors
- Variety of interface metaphors for reviewing the content structure

- Ability to edit raw material assets at a low application level
- Scripting language capability
- For template-driven tools, support for developer-generated templates
- Support for a sufficient number and flexibility of assessment types

- Includes robust programming features:
  - Accurate browser emulation
  - “Round-trip” editing of code
  - Launch source object editing applications from within the tool
  - No special non-standard or proprietary code used in conjunction with HTML
  - Set control parameters for media objects
- “What can the software do in a right-out-of-the-box, plug-and-play use case scenario?”  
**not** “What can the software do?”



# General Recommendations

- Avoid first releases of software.
- Ask vendor who clients are, what they use the tool for, and see if you can talk to them.
- Ask vendor for demo running on your enterprise system(s).
- Avoid companies with short history in the market.
- Buy only the components you need, if buying an LCMS.

- For a web-based tool, consider a hosted solution.
- Use tools in combination.
- Try the tool out on the system configuration your authors typically use.



# Current Trends in Authoring Tools

- Team-based life cycle production and maintenance
- XML Output
- Separation of content and appearance
- Centralized control with distributed contribution
- Complex templates and skins
- Learning object-centric architecture
- Embedded best practice design principles
- Support for advanced learning technologies

- For more information:
  - Peter Berking
  - 703.575.2017
  - [peter.berking.ctr@adlnet.gov](mailto:peter.berking.ctr@adlnet.gov)
- Slides and white paper are available at <http://www.adlnet.gov/News/articles/index.aspx?ID=522>
- Visit our new web site at [www.adlnet.gov](http://www.adlnet.gov) after March 31 launch.

