

Article

[Peter Cooper](#) · May 10, 2018 3m read

Replacing ZEN - Part 2 - Strategic Issues

[Index to Articles](#)

Published 2018-05-11 Last edit -

Hi All

In this article I detail some strategic issues that a new development UI will need to address - these are the ones that I can think of now - others may come to light during this journey.

See the webinar by Eduard Lebedyuk [here](#) from the last Global Summit describing modern web development and Caché

And, as always, if I have missed something please comment....

Here's my list (so far):-

- Choosing a Full Stack Development Environment With ZEN I need to know ooDesign, COS, Studio (for editing), COS, HTML, Javascript, CSS, FOP !!!!! And each of these is large in themselves. There is a good video [here](#) that gives an overview of the choices - but it's dated as it does not include containers But the detailed choices huge -The video [here](#) details hundreds of different technologies/tools/environments - it's awesome and frightening The key decisions I need to make are
 - What client development environment to use
 - What development environment (Caché Studio or Eclipse & Atelier - Which client editor to use)
 - What version control system that controls and moves both Caché classes and client files (CSS, JS, HTML or whatever)

As I said in part 1, I want to concentrate my intellectual effort to deliver complex business applications not the tools and always having to learn new stuff.

- Leverage Existing ZEN code (link under construction) Again, as in the introduction I would want to leverage/translate the existing ZEN definitions to use going forward. This will:-
 - Speed up development
 - Reduce testing - end-user acceptance testing is a large overhead for a small/medium company
- Sessions or No Sessions (link under construction) A key concept for JSON communication between the client and the server is that each interaction is self contained, ie the server and the client need to know nothing about each other over the content of each restful call and the JSON response - ie no CSP sessions Is this correct or do we need to maintain sessions?
- Security (link under construction) Using JSON - how do you implement a logon? What are the licensing issues? How do you prevent users hacking restful calls that they have no access to? There is a post by Anna Kantola [here](#) that describes some problems. logging in via rest
- Screen Layout/Responsive App (link under construction) As I said in the introduction, responsive apps for a desktop business application is not a high priority However it would be great to have a single framework that can be used for both static screens on the desktop *and* for responsive apps.
- Localisation (multi-lingual pages) (link under construction) ZEN/CSP has in-built localisation (csptext) that allows a single page definition to localise according to the browser language settings - it's a bit fiddly but works great allowing a single page to seamlessly appear in the user's preferred language . What other

options are there? Also date/time localisation.

Most of the above (will) link to a separate page to reduce the size of the article and to allow comments to be added to the appropriate heading

[Index to Articles](#)

[#Angular2](#) [#CSP](#) [#JSON](#) [#Frontend](#) [#ZEN](#) [#Caché](#)

Source URL: <https://community.intersystems.com/post/replacing-zen-part-2-strategic-issues>