



Xsemble

Visual Assembly of Software

Ashish Belagali

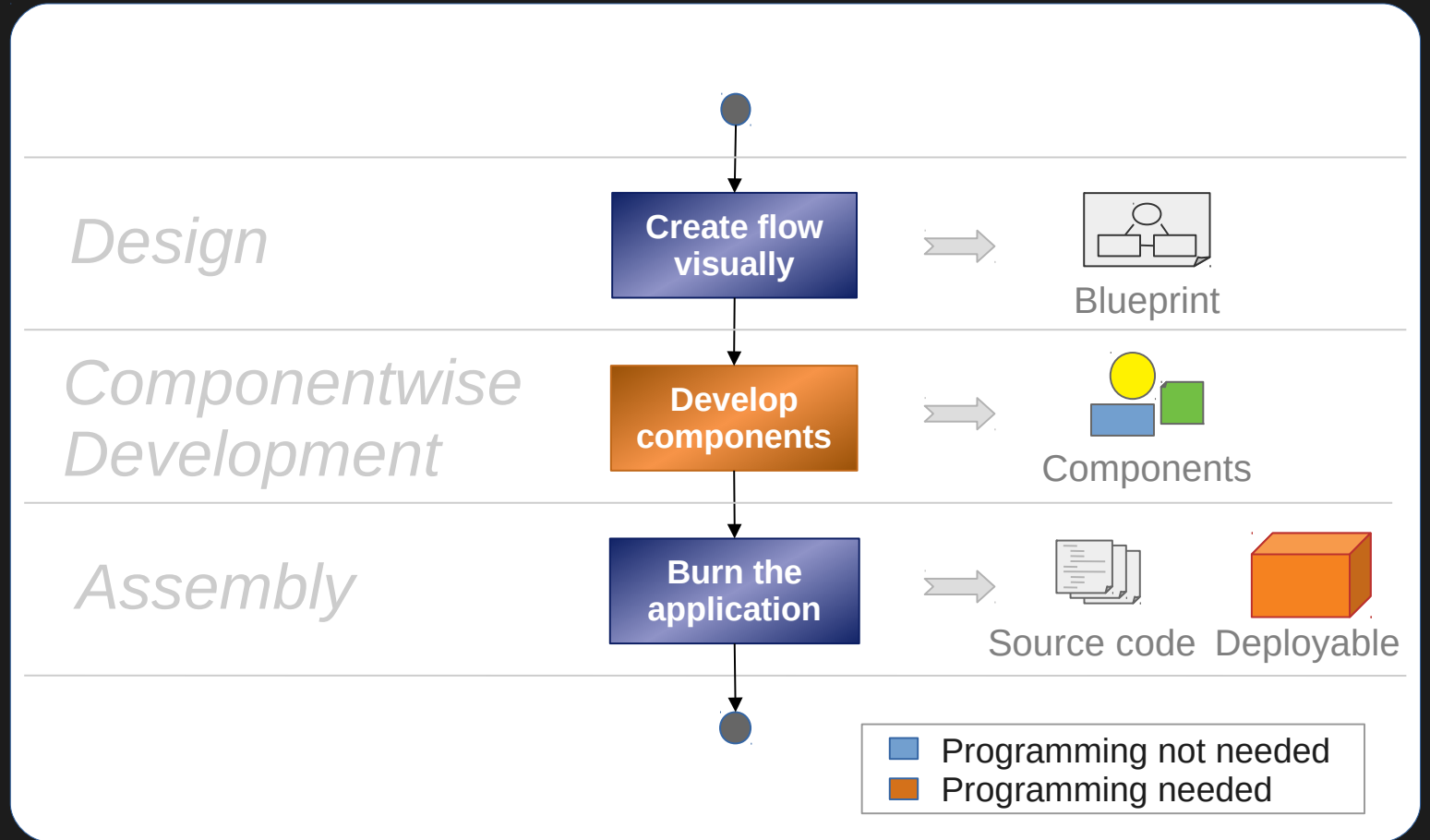
Email: ab@10Xofy.com

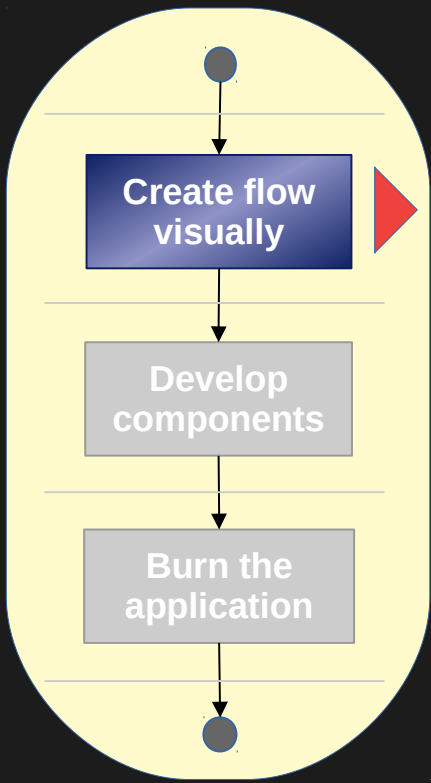
Cell: +91 98900 56365

Twitter @10Xofy

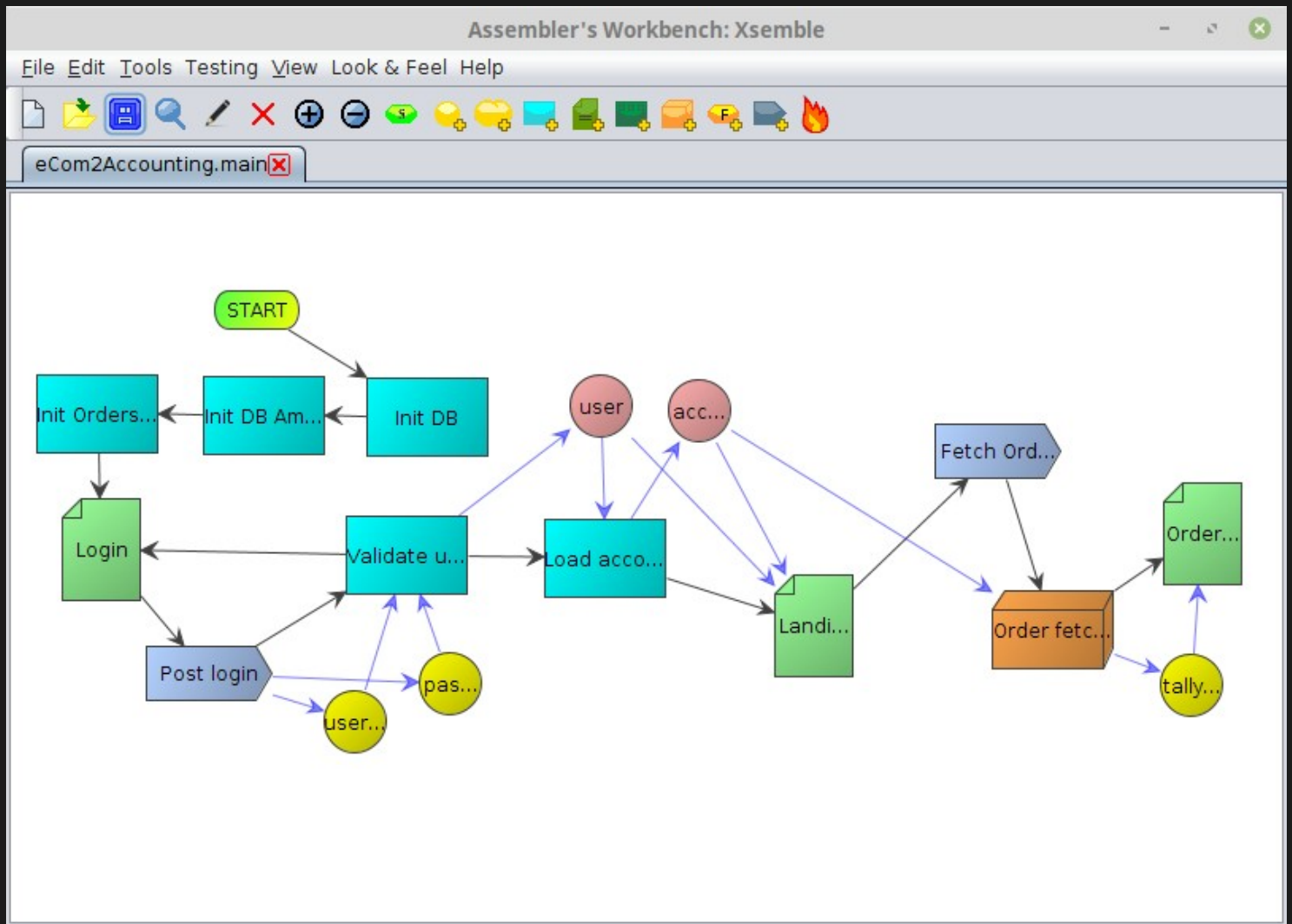
Make Software..

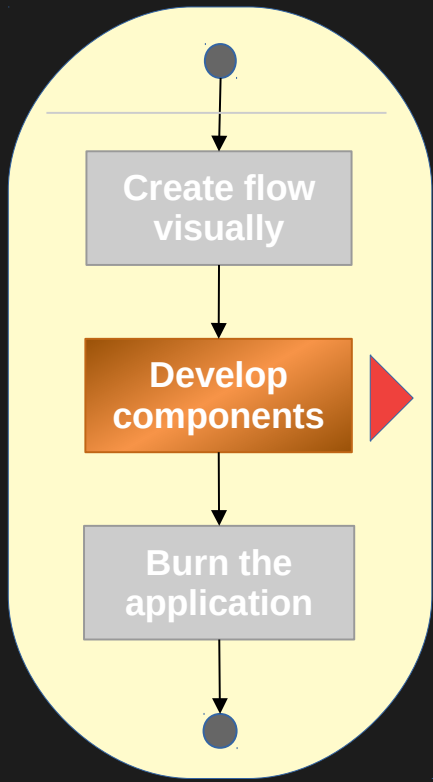
Like They Make
Cars!





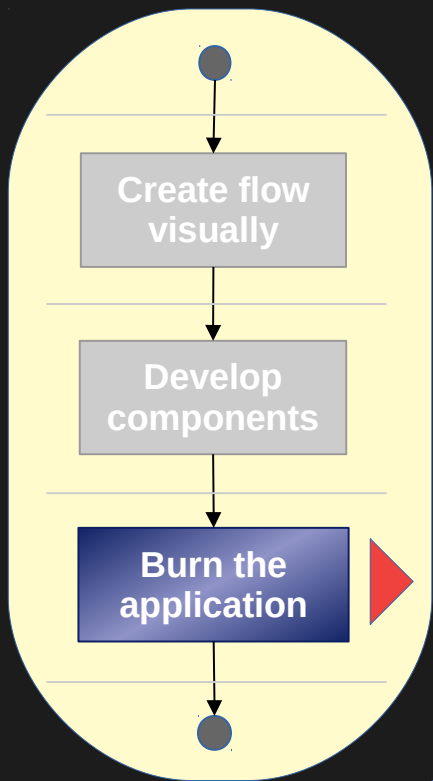
- Easy
- Intuitive



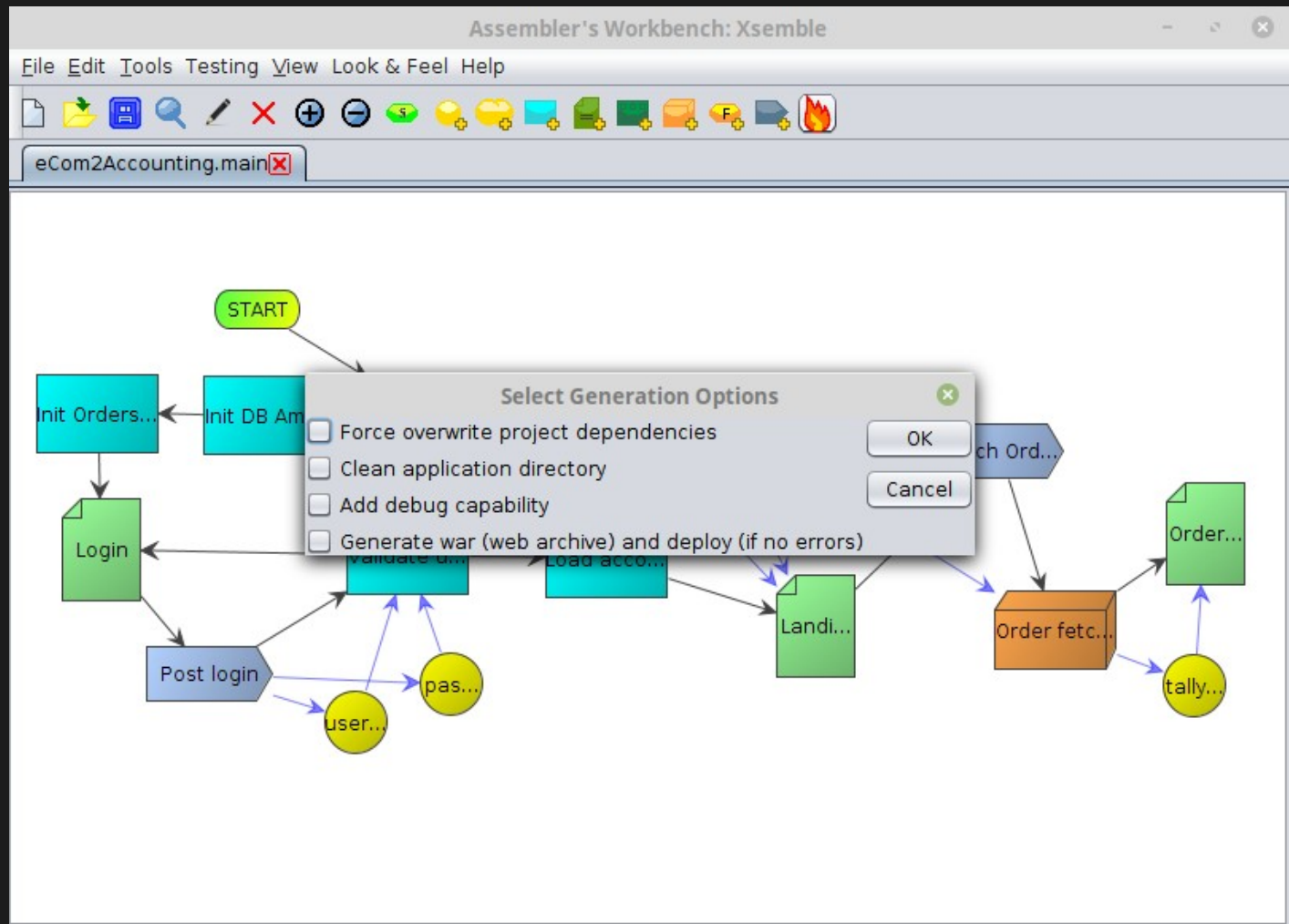


- From Generated Templates
- Type safe, Robust
- Smart logging
- Java language
- Use favorite editor

```
Add_Salutation.java (~)
File Edit View Search Tools Documents Help
Open Save Undo
Add_Salutation.java x
14 public class Add_salutation
15 {
16     private static final Logger logger = LoggerFactory.getLogger(Add_salutation.class);
17
18     public Map<String,Object> run(Map<String,Object> inargs)
19     {
20         logger.info("Session:"+ inargs.get("session")+ " - Entered method " + inargs.get("meth"));
21         // Get the in arguments
22         java.lang.String name_in = (java.lang.String) inargs.get("name");
23
24         // Out argument declarations
25         java.lang.String modified_name_out;
26
27         // Method exit path
28         String exitpath; // Possible values are "next"
29
30         // TODO: Your implementation comes here. Populate the appropriate values in the out arguments and exitpath.
31
32         // Create outargs and populate it
33         HashMap<String, Object> outData = new HashMap<String, Object>();
34         outData.put("_exitPath",exitpath);
35         outData.put("modified name", modified_name_out);
36
37         return outData;
38     }
39 }
40 }
Java Tab Width: 4 Ln 11, Col 4 INS
```



- 🔗 Create application sources
- 🔗 Optionally deploy
- 🔗 Glue code connecting components auto-generated – error-free, efficient
- 🔗 Application always sticks to the design



Live Monitoring

- Application deployed locally or remotely
- Aids in understanding and troubleshooting

The image shows two overlapping windows. The background window is a Mozilla Firefox browser displaying a web application at localhost:8080/Hello/x. The application has a form with the text "Enter name:" followed by an input field and a "Go" button. The foreground window is titled "Monitoring Workbench: Xsemble" and displays a flowchart of the application's execution. The flowchart starts with a "START" node, leading to a yellow "Ask name" node. From "Ask name", the flow goes to a "user" actor node, then to an "Entry Get Name" node, which leads to a "user name" data store node. Finally, the flow goes to a "Proce" (likely Process) node.

Advantage 1: Easy Development

The programmer's job is reduced to developing *a single module at a time* – as opposed to looking at the complete application.

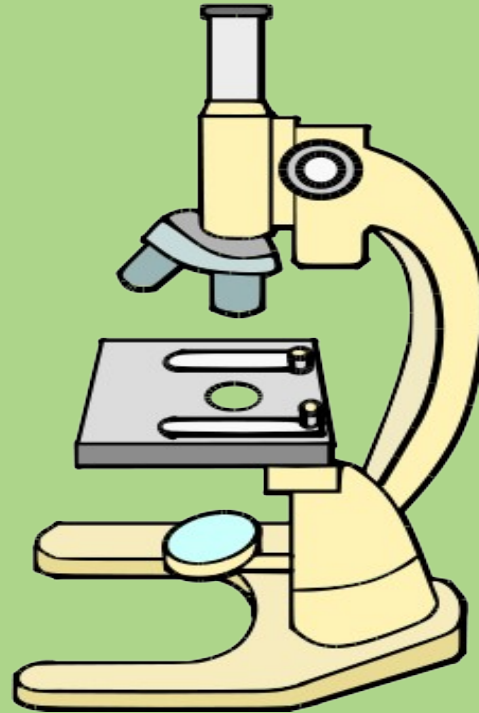


Benefits

- Less errors, Higher quality
- Skill requirement lower
- Efficient use of programmers – engage for one module at a time
- Better IP protection – No programmer has all the code

Advantage 2: Visibility

The visual design (aka blueprint) of the application is always current with the application.



Benefits

- Learning curve of new joinees reduced.
- Non programmers -- such as Managers, Domain experts, Support engineers – get to see the exact workflow and that helps in their jobs.

Advantage 3: Easy Maintenance

Source of an error can be isolated to a module *even before involving a programmer.*

Programmer performs a *surgical fix.*



Benefits

- ❖ Reduced effort and downtime
- ❖ A new programmer can handle it, as a single module is much easier to understand.

SDLC Impact

- Reduced effort
- Reduced risk

Cost saving \neq Effort saving.

Through efficient resource management and smart outsourcing, the cost saving could be even higher.

Waterfall or Agile – SDLC phases are a standard to build effort estimation.
Note: Actual saving depends on the engagement-specific situations.

SDLC Phase	Conventional	Xsemble	Reason for Reduction
Requirements	15	10	Technical resources are not needed in requirement analysis phase.
Design	25	20	Xsemble flow makes design explicit, which may be iterated upon. Xsemble's Health Check facility helps in design validation. Xsemble's sizing facility helps arrive at a reliable, granular estimation.
Development	40	20	Generated code templates, unit test templates, generated glue code reduce time. Major savings from reduced rework, as handling small code is orders of magnitude easier, less error-prone and less risky to handle than an application's large codebase. Xsemble's integrated project progress tracking aids in knowing the development progress.
Testing	15	7	Only integration testing, if unit testing is covered in development phase. Bugs are isolated to components and hence quicker to fix.
Deployment and User Acceptance	5	3	Any omissions / enhancements can be handled easily. Changes do not affect globally.
Total	100	60	

Maintenance Impact

Xsemble makes maintenance trivial. Maintaining components is easy and the complexity of maintenance stays low throughout the life of the software, as opposed to conventional way, where the software becomes exponentially complex and costly to maintain with time.

Downtime and errors introduced are reduced dramatically.

Software life increases.

Both these factors together mean a good saving of the maintenance cost.

SDLC Phase	Conventional	Xsemble	Reason for Reduction
Maintenance	100	20	Programmer is engaged on need basis after the support team identifies defective module. The programmer needs to go through the code of only the defective module, without worrying about the rest of the code.
Total	100	20	

Because of these factors, it may be a good call to invest in migrating software in maintenance phase to Xsemble, than to keep maintaining it in conventional way.

Case Study 1

Pilot outside our company

Xsemble was piloted outside our company for the first time at SoftDEL in September 2017.

SoftDEL is a leading IoT company and works in the domain of Building Automation. It got a prestigious opportunity to work with one of the top companies in the world to develop a part of the Building Management Solution. Xsemble (it was called MGF then) was used in creating the first POC delivery in record time.

Challenges:

- ❖ The delivery timeline was extremely short (~15 days) and was immovable.
- ❖ A team was freshly recruited for the project. It was completely new to the domain. SoftDEL employees did a great job to impart necessary training in a short time.
- ❖ The team comprised of 2 freshers just out of a Java training, led by a senior consultant who also knew Xsemble. Traditionally, such freshers are not considered productive.

How it went:

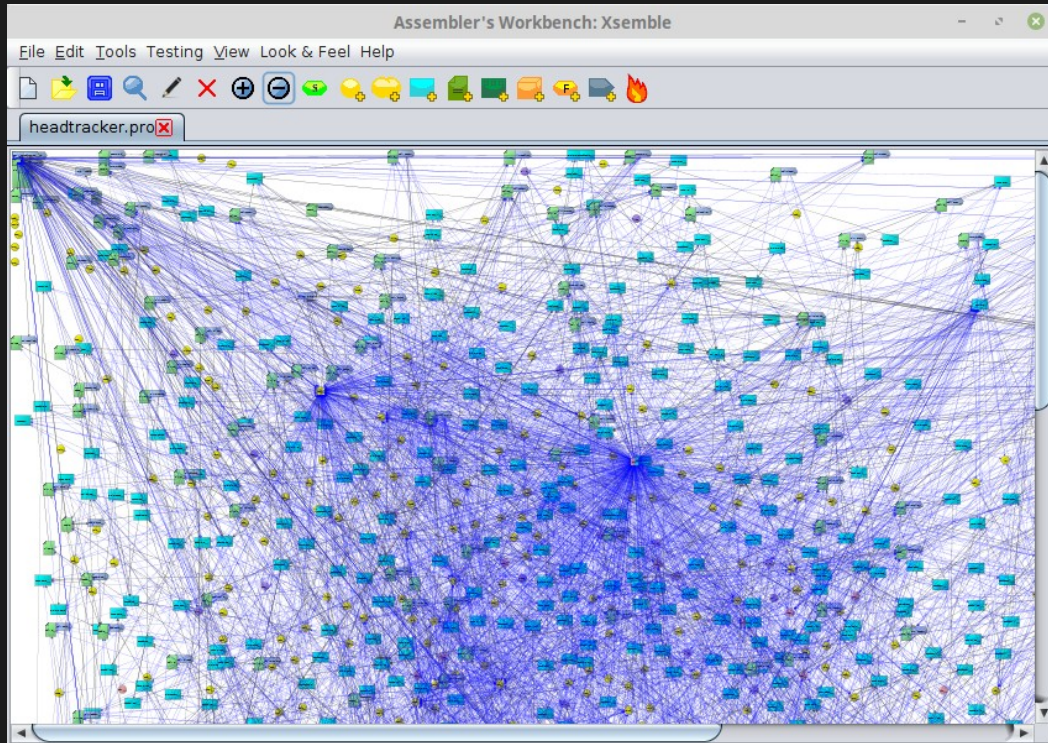
- ❖ Delivery of two web applications, dockerized, was made in time, successfully!
- ❖ Xsemble graphs made the working and the progress visible to the SoftDEL stakeholders, which helped especially as the project was critical.
- ❖ The freshers were quick to grasp Xsemble. While they could not write servlets on their own, they were productive with Xsemble within a very short time.
- ❖ SoftDEL said it got 30-32% cost saving due to Xsemble.

Case Study 2

Complex application

Xsemble was used to create a full-featured recruitment application – HeadTracker.

HeadTracker handles with ease an ever growing candidate database (40000+, last checked) for one of the customers.



Sample functionalities (just to give an idea of the depth):

- 6 ways of candidate addition – resume parsing, excel import, email import etc.
- Client and requirements creation.
- Proposal tracking – from proposing to evaluation steps (interviews, GD etc) to accept/reject
- Manual and auto-alerts (trigger on proposal state change)
- Flagging of candidates with flags getting custom meaning
- Trash can and undelete functionality
- Custom fields of candidate – Search includes these fields and flags too.

▶ The blueprint of HeadTracker is a single Xsemble project consisting of 1231 nodes. (It is a single project because the subproject functionality of Xsemble was not ready then.)

HeadTracker components are 138 KLOC of source code (not counting Xsemble generated code).

Case Study 3

Xsemble for Education

Xsemble component development turns out to be a great way to train students on the practical aspects of programming..

SICSR (Symbiosis Institute of Computer Studies and Research) is a leading IT college from Pune. 10Xofy offered mass internship projects to its students. The content of the internships was to work on individual components exported out of Xsemble (for a live project). 10Xofy offered help with the problem statement, but mandated that students will have to figure out the technology on their own.

Challenges:

- ❖ IT graduates are not deemed as job ready, 95% of them incapable of coding. The main roadblock is lack of practical understanding.
- ❖ The syllabi from various institutes do not emphasize nor measure practical learning, creating graduates with degrees but of little use to the industry.
- ❖ By and large, students lack a context to learn, guidance and a starting point.

How the mass internships went:

- ❖ The initial reaction of students was that the task is out of their reach.
- ❖ However, a good faculty exerted sternness and offered help at the same time. She sat with students to figure out the problem.
- ❖ Soon the students picked up steam and started delivering. They were elated when the components created by them were accepted and that motivated them further.
- ❖ The knowledge acquired through self-learning and the confidence acquired through the component acceptance made the students ready to tackle industry challenges.

Use Scenarios

Startup
founders

Product
owners



Xsemble

Visual Assembly of Software

Budget
projects

Strong IP
projects

Next Steps

Let's engage!

We offer to have engagements with deep involvement aimed at ensuring that business benefits are realized!



Connects

Ashish Belagali

Email: ab@10Xofy.com

Cell: +91 98900 56365

10Xofy

Web: <http://10Xofy.com>

Twitter @10Xofy

LinkedIn: 10Xofy

Xsemble channel

FB: fb.me/Xsemble

YouTube: [xsemble](https://www.youtube.com/channel/UCxsemble)