L4 Project Supervision Workshop

2 September 2019, SAWB 423

This document provides a brief summary of interesting ideas shared at the Level 4 project supervisors workshop. The event had two group discussions, one on the project timeline and one on supervision strategy, and a short panel gathering insights from experienced supervisors.

These constitute personal recommendations. Every project is supervised differently, and ideas may or may not be applicable to some types or styles of projects.

Timeline strategies

Experiences of timelines. Most supervisors felt students started fast in Week 1, but then struggled to make much progress until Week 10 when coursework wound down. Often impressive progress in Week 10-13, but this structure still led to heavy pressure in Semester 2 to catch up.

Accelerate initial learning. Students often have a long lead time reading tutorials and getting to grips with technology. Encourage students firmly to get this underway over the summer, and use Week 1, when no classes are scheduled, to really get basic concepts under control.

Work around resource constraints and plan early for this. Some projects need special hardware, like robots, VR headsets or compute clusters. This will lead to inevitable crunches in the dense weeks just before Christmas. This needs to be managed fairly and organised early to avoid stress and unhappiness. Students need to be aware that their time with these devices is limited and must plan ahead for use!

Make sure students have time for evaluation. Students often want to code until the end of the project, and end up either neglecting evaluation, or evaluating something different from the final product because they have modified it post-evaluation. Encourage early stopping in development, for both evaluation and writing.

Prototype by Christmas. Several supervisors suggested their students have a working prototype with the bulk of features before the Christmas break. Ideally some code running by Week 7-8 and an initial version by week 10. In reality this can slip to the following semester but this is a warning sign...

Dissertations: Little and early. Ask students to start writing the dissertation *from Week 1* -- and be willing to show and get feedback on incremental drafts every week. Explain that much of the early work may be discarded but there is value in writing it anyway, for feedback on style, structure, etc. Perhaps ask for a table of contents by approximately week 6.

Provide multiple timeline examples for students. L4 coordinator could (should) provide a number (2-3) skeleton timelines that cover different styles of projects/supervisors to cover the diversity of project configurations, for both supervisors and students to look at.

Publish coursework deadlines to supervisors. Students often complain they have too much coursework to make much progress, which they often do -- but supervisors are often out of the loop about the current coursework levels. It would be helpful for the L4 project coordinator to publish the timeline of coursework load so supervisors can see how the current coursework loads.

Thoughts from experienced supervisors

Request formal requirements early on (e.g. in first few weeks), so that there is a shared understanding of what must, should, could be implemented between student and supervisor.

Remind students that L4 projects do not have to be original research work (though they can be, and this can be motivating).

Consider introducing L4 students to the entire research group, and integrating them thoroughly with early meetings including RAs, PhD students and other researchers. Give them context about the value of their work. Add them to research group lists and make them feel part of the team.

Have students present at the whiteboard and draw out their ideas in front of you.

Talk about risk early, and avoid students putting off risky/unknown things to the end of the project. Prioritise risky parts if possible.

Devote equal time to strong students, and find ways to push them forward even if it is tempting just to let them continue on.

Collaborative projects with real clients (e.g. across university) can be very rewarding but have large risks. Tension between collaborators development needs and students' educational needs; supervisors taking on such projects need to be willing to step in to make sure student interests are represented. Things can go really wrong otherwise!

Meeting strategies

Actively chase students who don't attend meetings or provide something (code, drafts, etc.) that you expected. If students don't have anything to talk about, use the time as an opportunity to reflect on the project and reassess where it is going, rather than the temptation of just finishing up early.

Hold back-to-back meetings in regular slots to keep students on schedule and avoid overrunning meetings (and also underrunning meetings).

Use Slack or other project management tools to rapidly respond to queries and avoid students getting stuck on trivial issues.

Keep free days free. If students have a whole day block without classes, don't schedule meetings on that day. Instead sandwich meetings around other classes and let students keep solid blocks to work on project/coursework ideas.

Cross-supervisor support sessions. Consider how to hold sessions across supervisors to support students early on, e.g. setting up robots, getting a hello world on a VR headset, configuring compute clusters and issuing jobs, mastering basic ML libraries, setting up a common research environment, etc. A small coordination effort across a few supervisors can save a huge amount of time.

Group meetings Can work well to share knowledge and keep students at a common pace but have risks in terms of personality effects (disadvantaging shy students) and can make it hard to honestly criticise work. Also hard to administer pastoral care in a group environment (a key supervisor role)

Alternate group/individual meetings (one week group, one week individual) were proposed as a successful way to marry the benefits of group meetings with individual supervision.

Frequent, messy drafts. Students should be made comfortable submitting early drafts, even if messy or scrappy and getting feedback as early as possible.

Two-in-a-row. More than one consecutive meeting missed is a warning sign that things are possibly going very wrong. Don't let students miss more than one meeting.

Advance agenda. Have all students submit an agenda for the supervision meeting 24 hours in advance, and run through the agenda in the meeting.

Three slides. Ask students to prepare three slides and present them at each meeting: what have you done, what questions do you have and what is your plan?

Code checkout. Ask students to keep all code under version control, and live checkout, compile and execute in the meetings to keep students “honest”. Likewise, suggest students (in less code based projects) work on a single master document that supervisors have access to, and can see any updates as they happen.