

LUA_TE_X 1.10, a stable release

Na wat eerste experimenten is het LUA_TE_X project in 2006 van start gegaan. In eerste instantie waren Taco Hoekwater, Hartmut Henkel en Hans Hagen betrokken, later kwam daar Luigi Scarso bij terwijl Hartmut wat uit het zicht verdween. Vanaf het begin zijn de CON_TE_XT gebruikers betrokken bij het project, niet in de laatste plaats omdat ze gewillige testers waren. Via `contextgarden.net` waar Mojca Miklavcic de regie heeft, onder andere over de `compile farm`, kunnen gebruikers distributies downloaden met de meest recente features. Het is dus eigenlijk een in Nederland geïnitieerd project van de CON_TE_XT ontwikkelaars en gebruikers.

Een paar jaar terug is versie 1.0 van LUA_TE_X gelanceerd: in principe was de interface stabiel. Binnenkort lanceren we versie 1.10, met wat kleine uitbreidingen en natuurlijk verbeteringen en fixes. Dit is een verdere stap naar stabilisering. Omdat ook andere macro pakketten gebruik gingen maken van LUA_TE_X werd duidelijk dat verder experimenteren geen echte optie meer was, meer daarover later.

Samengevat biedt LUA_TE_X het volgende:

- Een up-to-date T_EX engine, met een volledig UTF-8 code pad. De input is UTF, de logging ook, en karakters en shapes (glyphs) maken gebruik van de in UNICODE toegestane ranges.
- Een open font interface die het mogelijk maakt zowel traditionele 8 bit fonts (meestal zijn dat TYPE1 fonts) te gebruiken maar ook OPEN_TY_PE fonts. En dat alles onder volledige controle van T_EX en LUA.
- Een extensie interface die het via zogenaamde LUA callbacks mogelijk maakt de input te manipuleren, zelf extra commandos te definiëren (met behulp van scanners), en de tussenresultaten, in de regel zogenaamde node lists, te onderscheppen en manipuleren.
- Dat alles wordt gedaan in LUA, een relatief eenvoudige maar krachtige scripting taal, met een stabiele reputatie en geen afhankelijkheden van grote, steeds wijzigende en uitdijende bibliotheken.
- Een bewezen stabiel PDF backend dat efficiënt ondersteuning biedt aan uit PDF overgenomen uitbreidingen als font expansie maar ook kan omgaan met de uit OMEGA overgenomen mogelijkheden om bijvoorbeeld van rechts naar links te typesetten.
- Images zijn gepromoveerd naar een standaard functionaliteit. Ook is er een interface naar het image subsysteem.
- Er zijn in de loop der tijd wat extra primitieven bijgekomen (vooral handig voor ontwikkelaars) terwijl wat PDF_TE_X extensies zijn verdwenen omdat we LUA hebben als vervanger.
- De codebase is destijds door Taco overgezet van PASCAL naar C en is in de loop der tijd uitgebreid en opgeschoond. De engine is efficiënt en goed te onderhouden. De performance is ok en als men LUA gebruikt kan men de LUA_JIT variant kiezen die een snellere virtuele machine heeft.

Kortom, na meer dan een decennium is er een T_EX engine die zijn werk goed kan doen en die eigenlijk vanaf het begin (in ieder geval door CON_TE_XT gebruikers) gewoon in productie kon worden gebruikt.

Stabiliteit is belangrijk binnen de T_EX wereld en daarom is versie 1.10 min of meer het eindpunt van een ontwikkeling. Natuurlijk komen er updates, maar net zoals PDF_TE_X al jaren stabiel is in functionaliteit, zo zal ook LUA_TE_X dat moeten zijn.

Echter, omdat we nog wel wat ideeën hebben, gaan we wel door met experimenteren. Dit is binnen de `CONTEXT` community relatief eenvoudig. Er is bijvoorbeeld een bewezen effectieve infrastructuur die dat faciliteert. Daarnaast hebben we niet te maken met gebruikers (denk daarbij aan instellingen en uitgevers) die helemaal geen verandering willen en zelfs structurele aanpassingen in een macro pakket niet toejuichen. In `CONTEXT` is dat zelden een probleem omdat we een consistente interface hebben en veel is geïntegreerd. De transitie van `MkII` naar `MkIV` is redelijk soepel verlopen: de belangrijkste wijzigingen voor gebruikers hadden te maken met encodings en fonts.

Een follow-up van `LUATEX`, denk aan `LUATEX` versie 2+, zal dan ook wederom binnen het `CONTEXT` domein plaatsvinden. Op die manier heeft niemand anders er last van. Een beetje zoals in de eerste jaren van `LUATEX`, toen alleen `CONTEXT` er gebruik van maakte en we drastisch konden experimenteren. Ik herinner me nog de tijden dat de binaries en `CONTEXT` code dagelijks heen en weer gingen tussen Taco en mij, waarbij we ideeën konden uittesten. Ik heb daar prettige herinneringen aan.

We kunnen sommige aspecten van een follow up terugsluizen naar `LUATEX`. Standaard zal `CONTEXT` deze engine gebruiken en we zijn dus gebaat bij consistentie (in fixes) en functionaliteit, maar omdat we geen stabiliteit willen doorbreken is er dus gewoon een extra engine (die waarschijnlijk alleen door `CONTEXT` wordt gebruikt). De werktitel voor dit project is `METATEX`, een naam die al langer binnen het project wordt gebruikt, en de extra engine zal de naam `luametatex` hebben zodat er geen verwarring ontstaat met de standaard `luatex` engine. De `CONTEXT` gebruikers zullen voornamelijk gewoon `LUATEX` gebruiken, maar kunnen dan als alternatief experimenteren met de follow up.

Gedurende de `LUATEX` ontwikkeling hebben we steeds bijgehouden wat we deden en er frequent over gerapporteerd, op papier en tijdens bijeenkomsten. Er zijn inmiddels vier documenten in de `CONTEXT` distributie die de stadia en ontwikkelingen daarbinnen samenvatten. Een vijfde document dat de follow up bijhoudt ontstaat op het moment van dit schrijven.

Hans Hagen