

Löytöretkiä sähköiseen MATEMATIIKKAAN

Tervetuloa!

Olen **Petri Salmela Four Ferries Oy**:stä.

Keitä olemme?

Olemme tutkineet yliopistolla ja kehittäneet matematiikan **sähköistä esittämistä, lukemista ja tekemistä**.

Tutkimusta tehtiin alkujaan **Ralph-Johan Backin** johtamassa tutkimusryhmässä **Åbo Akademin** ja **Turun yliopiston** IT-laitosten yhteisessä resurssikeskus IMPEdissä useammassa projektissa. Nyt olemme siirtäneet kehityksen yritykseen, jonka on tarkoitus **kaupallistaa** tuloksia ja tuotoksia, etteivät hautaudu pöytälaatikkoon.

Projektit

- OPTEK
- Emath
- Emath 2
- Virum
- Automaattinen tarkistaminen

Ja lopulta siirtyminen 4Ferriesiin.

OPTEK

Yläkoulun materiaaleja ja pilotointeja. Sähköisen materiaalin lukemista (Moodle, www-sivut), kirjoittamista (LyX), ratkaisujen tarkistamista (LyX -> PVS)

Ei merkittävästi eroa periteisesti opiskelleeseen vertailuryhmään.

Emath

EU-rahoitteinen projekti, johon osallistuivat

- Turun kaupunki
- Tukholman kaupunki
- Tallinnan kaupunki
- Ahvenanmaan maakuntahallitus (Maarianhamina)
- Turun yliopisto
- Åbo Akademi

Lisäksi Rasion lukio osallistui pilotointiin.

Yliopistot loivat sähköisen opiskeluympäristön ja materiaalin.

Lukion pitkän matematiikan ensimmäisen vuoden materiaali suomeksi ja ruotsiksi, sovitukset vastaavalle tasolle Ruotsin ja Viron opetusohjelmiin.

Kaupunkien kouluissa pilotoitiin järjestelmää ja materiaaleja.

Emath 2

Projektissa käytettiin Emath-projektin järjestelmää ja kirjoitettiin valmiiksi loput pitkän matematiikan kurssikirjat sekä päivitettiin ne uuteen opetussuunnitelmaan.

Virum

Ruotsinkieliset lukiot kaipasivat yhteistyöelintä, jonka kautta voisivat pitää yhteisiä virtuaalikursseja. Erityisesti kursseilla, joissa osallistujamäärät olisivat yksittäisessä koulussa niin pieniä, että kurssien toteuttaminen ei yksin olisi mahdollista.

Toteutettiin Åbo Akademin ja Four Ferries Oy:n yhteistyönä edellisten projektien opeilla sähköinen opiskeluympäristö.

Automaattinen tarkistaminen

Muiden projektien rinnalla oli koko ajan automaattisen tarkistamisen projekti, jossa tutkittiin, miten matemaattiset ratkaisut voidaan kirjoittaa rakenteiseen muotoon ja tarkistaa automaattisesti teoreemantarkistimella.

Automaattinen tarkistaminen mahdollistaa välittömän palautteen antamisen opiskelijalle. Tarkoitus ei ole korvata opettajaa vaan täydentää.

Kysymyksiä

- Miten matematiikkaa voitaisiin kirjoittaa sähköisesti?
- Miten opiskelijat lukevat ja kirjoittavat sähköisesti ja miten eroaa perinteisestä?
- Miten ratkaisuja voisi tarkistaa automaattisesti?

Ralph-Johan Back ja Jockum von Wright ovat kehittäneet Åbo Akademiilla matemaattisten todistusten ja laskujen esittämiseen rakenteisten päättelyketjujen menetelmän ja projekteissa käytettiin yleensä tätä laskujen esittämiseen. Tämä on ollut myös pohjana automaattisen tarkistimen rakentamiselle, sillä automatiikka tarvitsee selkeän rakenteen, jotta se pystyy ymmärtämään, mitä ollaan tekemässä.

Tutkimusprojektit ovat olleet joukko löytöretkiä aiheeseen ja seuraavalle retkelle on aina lähdetty edellisen opein. Paremmin ymmärtäen, mitä ollaan etsimässä ja tutkimassa.

Seuraavassa joitakin löytöretkien löytöjä tai oppeja.

Web!

Web-ympäristö on käytännössä järkevin / ainoa järkevä alusta tällä hetkellä.

Toimii periaatteessa kaikilla käyttöjärjestelmäalustoilla, mahdollista upottaa melkein mihin tahansa ympäristöön ja tarvittaessa itsenäiseksi sovellukseksi.

Matematiikan esittäminen webissä on perinteisesti ollut ongelmallista.

Verrataan muihin aineisiin. Esimerkiksi biologiassa pääsee pitkälle jo pelkällä tekstillä ja kuvilla. Äänillä, interaktiivisilla kartoilla yms. rikastettu sisältö on tietenkin monipuolisempaa, mutta tekstillä ja kuvilla pääsee hyvin alkuun.

Matematiikassa sen sijaan törmätään heti kaavoihin. Kaavoissa on sisäistä rakennetta, joka esitetään usein erikoisilla merkeillä ja asettelulla, jonka ladonta ei ole helppoa. Lisäksi tarvitaan erilaisia kuvaajia ja kaavioita.

Tietotekniikka on matematiikan lapsi, mutta tässä suhteessa

"Suutarin äidillä ei ole kenkiä"

LaTeX

Yksi suuri syy on nähdäkseni LaTeX, joka tekee todella laadukasta jälkeä ja joka on yliopiston tutkijoille yleensä helppo ja riittävä. Ei ole ollut tarvetta tehdä helpompaa.

Nyt kuitenkin opetuksen sähköistymisen myötä matematiikka pitäisikin pystyä kirjoittamaan sähköisesti myös lukiossa ja yläkoulussakin. LaTeX ei tähän taivu, koska vain hyvin pieni osa opiskelijoista lopulta päätyy opiskelemaan matematiikkaa tai fysiikkaa. Ja heistäkin aika moni näyttäisi mieluummin klikuttelevan Wordin kaavaeditorilla.

LyX

LyX on kätevä välimalli. Tekstinkäsittelyohjelma, jossa matematiikka on luonnollinen osa kirjoittamista ja näytetään LaTeX-tekstin sijasta oikean näköisinä kaavoina. "Ensimmäisen luokan kansalainen"

LyXiä käytettiin OPTEK-projektissa laskujen laskemiseen. Erillisenä ja monipuolisena ohjelmana vaati kuitenkin asentamisen ja oman opettelunsa. Automaattista tarkistamista ajatellen hankala.

Datana

Alkujaan matemaattiset kaavat esitettiin webissä kuvina. Kuvien käyttö on kuitenkin pidemmän päälle hankalaa. Eivät reagoi hyvin skaalaamiseen, fontin vaihtoon, kopiointiin, tausta- ja tekstivärien vaihtoon tai automaattiseen tarkistamiseen.

Parempi, mitä enemmän sisältö tallennetaan rakenteisena datana. Tällöin sisältö voidaan tarvittaessa esittää erilaisilla esitystavoilla tai syöttää koneelle prosessoitavaksi taikka tarkistettavaksi.

Sama pätee tietenkin kaavojen lisäksi myös muuhun sisältöön, esimerkiksi kuvaajiin, geometriakuviin tai merkkikaavioihin.

OPTEK-projektissa tehdyssä materiaalissa kaavat olivat LaTeX-tekstiä, joka näytettiin jsMath-kirjastolla. jsMath kehittyi myöhemmin MathJaX:iksi, joka tekee upeaa jälkeä.

Automaattinen tarkistaminen tarvitsee rakenteista dataa. Yksittäisille kaavoille ei osata tehdä mitään vaan pitää olla jonkinlainen konteksti ja kytkös kaavojen välillä, jotta kone tietää, mistä on kyse.

Muokattavaa

JsMath ja MathJaX osasivat näyttää matematiikan webbisivulla hyvin, mutta kirjoittaminen tapahtui aina LaTeX-kielellä ja vaati osaamista. Ja sama pätee usein myös kuvaajiin, kaavioihin yms.

Matemaattisesta sisällöstä pitää saada helposti muokattavaa.

Päätelyketjueditori

Ensimmäiset prototyypit päätelyketjueditoreista olivat aika karuja ja selvästi prototyyppisiä.

Kaavat syötettiin tekstikenttiin LaTeXina ja näytettiin jsMathilla. Suunta oikea, mutta edelleen piti osata LaTeXia vaikka käytössä olikin symbolipaneeli. Jos jostain puuttui aaltosulje, ei kaava tullut näkyviin.

MathQuill

MathQuill oli lopulta ratkaiseva löytö. Sama kaavansyöttökenttä, jota YTL käyttää Abittiympäristössä. Tuo LyXin kaltaisen kaavan editoinnin web-ympäristöön. Voi kirjoittaa LaTeX-komennoilla tai symbolipanelia hyödyntämällä. **Kaava näytetään heti lopullisessa muodossaan.**

Yksittäisen kaavan syöttämisen lisäksi MathQuill mahdollistaa monimutkaisempien työkalujen rakentamisen, joissa kaavakenttä on komponenttina. Esimerkkeinä päätelyketjueditori ja yhtälöympäristö.

Työkaluja

MathQuillin innoittamana rakensimme useita eri sisältötyyppejä toteuttavia työkaluja, joilla kaikilla oli **katselutila** ja **muokkaustila**.

- Tekstielementti
- Kuvaelementti
- Yhtälöympäristö
- Päätelyketju
- Funktiopiirturi
- Geometriakuva
- Matematiikkataulukko
- Merkkikaavio
- Kulkukaavio

Kunkin tyyppin sisältö tallennetaan rakenteisena datana, joka on helposti muokattavissa ja tarvittaessa näytettävissä erilaisessa muodossa.

Emath-kirjat

Näiden työkalujen avulla luotiin Emath-projektin matematiikan kirjat. Kirja koostui sivuista ja sivut erilaisia sisältötyyppejä olevista elementeistä.

Kirja toimi opiskeluympäristönä, joka oli käyttäjien omalla koneella. Tehtävät tehtiin suoraan kirjaan ja ne palautettiin palvelimen kautta opettajan kirjaan, josta opettaja pääsi niitä tarkastelemaan.

Ratkaisuja

Matematiikkaa kirjoitettaessa ei yleensä haluta kirjoittaa vain yksittäisiä kaavoja vaan niistä koostetaan yleensä kokonaisuuksia. Ratkaisuja. Laskuja tai todistuksia.

Laskuja ja todistuksia

Laskussa ja todistuksessa ei ole vain yksittäisiä kaavoja vaan kaavoja, jotka ovat jossain relaatiossa keskenään. Keskenään yhtäsuuria tai ekvivalentteja, pienempiä tai suurempia kuin toinen, yksi on seurausta toisesta jne.

Tämän takia meidän mielestämme on tärkeää, että näille rakenteille on oma tietorakenteensa ja työkalu, jolla näitä kirjoitetaan. Käyttäjän ei tarvitse pohtia kaavojen sijoittamista ja esityksen sujuvuutta vaan työkalu antaa valmiin rakenteen, johon vaiheet syötetään. Uusi rivi valmiine kenttineen syntyy enter-näppäimen painalluksella. Myös tehdyn askeleen selitykselle tai perustelulle on oma paikkansa.

On makuasia, haluaako käyttäjä käyttää ratkaisunsa kirjoittamiseen perinteisen näköistä asettelua vai rakenteista päättelyketjua.

Tarkistin

Kun laskut on tehty yksittäisten kaavojen sijasta rakenteisina kokonaisuuksina, on ne mahdollista antaa koneen analysoitaviksi ja kone voi antaa yksityiskohtaista palautetta kertoen, mitkä ratkaisun vaiheet on laskettu oikein ja mitkä kohdat kannattaa laskea uudelleen.

Kokonaisuuksia

Vaikka laskun tai todistuksen esittäminen kokonaisuutena on jo hyvä askel, on hyvä ratkaisu usein kuitenkin enemmän kuin vain suora lasku. Se saattaa koostua osista, joita voivat olla esimerkiksi:

- Tehtävänantoa erittelevä teksti
- Tilannetta kuvaava piirros tai muu kuva
- Lasku tai useampia
- Kuvaaja tai kaavio
- Yhteenveto tekstinä
- Vastaus

Emath-ratkaisut

Emath-kirjoissa opiskelijoiden ratkaisutyökalu oli juuri kokonaisuuden osista koostava väline.

Laskut voitiin kirjoittaa joko päättelyketjuina tai yhtälöympäristöinä.

4f Vihko

Emath-projektien jälkeen lähdettiin kehitystä viemään entistä enemmän tähän suuntaan ja tuloksena oli yrityksessä kehitetty 4f Vihko, joka on HTML5-sovellus, jolla voi tehdä oman vihon. Vihko koostuu peräkkäin olevista sivuista, sivu koostuu eri tyyppisiä sisältöelementeistä. Kun vihkossa siirrytään muokkaustilaan, kaikki sivun sisältöelementit näytetään muokkaustilassa ja niiden väleihin voi lisätä uusia elementtejä paletista raahaamalla.

Katselutilassa sisältö on kohtuullisen miellyttävää katseltavaa.

4f Studio

Virum-projektissa opiskeluympäristö, 4f Studio, toteutettiin 4f Vihon ympärille. 4f Studion päänäkymässä näkyvät käyttäjän omat muistiinpanovihot, hänen koneelleen lataamat vain-luku-tilassa olevat kirjat (esimerkiksi Emath-kirjat) sekä kurssit, joihin hän on liittynyt.

Kun opettaja luo kurssin, hän valitsee kurssin materiaaliksi jonkin kirjan tai oman vihkonsa. Kurssi näkyy opiskelijalle näkymänä, jossa pääosassa on kyseinen materiaali ja reunassa erilaisia työkaluja. Ensimmäisenä työkaluna on viestintäkeskus, jolla opettaja ja opiskelijat viestivät keskenään ja opettaja merkitsee kotitehtävät. Muita työkaluja ovat mm. tilastotyökalu omien tai opiskelijoiden tehtävien teon tarkastelemiseen sekä muistiinpanotyökalu alleviivausten tekemiseen.

Opettaja voi tehdä materiaaliin muutoksia kurssin aikana, opiskelija luonnollisestikaan ei.

4f Studio toimii samalla periaatteella kuin aiempi Emath-ympäristö, eli materiaali on käyttäjän omalla koneella ja vain muutokset, viestit tai tehtävien ratkaisut lähetetään palvelimen kautta asianosaisille.

Seuraavaksi

Jatkotutkittavaa:

- Automaattisen tarkistamisen kehittäminen entistä kattavammaksi
- Voisiko muita sisältöjä, kuten geometriakuvia, analysoida automaattisesti?
- Sisällön tekijöitä kiinnostaa usein yhteistyö verkon yli. Esimerkiksi projektityöt.
- Mitä muita sisältötyyppejä kaivataan? Fysiikan voimakaaviot, kemian rakennekaavat,...
- Yhteistoiminta muiden ohjelmistojen kanssa. 4f Vihko on Abititikulla ja osan elementeistä voi jo nyt raahata suoraan Abitin vastauslaatikkoon järkevällä lopputuloksella.

Kiitos!

Kalvot, esityksen muistiinpanoja ja paljon kokeiltavaa löytyy osoitteesta:

<http://fourferries.com/itk2018/>