

Ville-Pekka Ranta-aho

Muistipalatsin proseduraalinen luonti

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Ohjelmistotekniikka

Insinööriytyö

2.12.2016

Tekijä(t) Otsikko	Ville-Pekka Ranta-aho Muistipalatsin proseduraalinen luonti
Sivumäärä Aika	28 sivua + 1 liite 2.12.2016
Tutkinto	Insinööri (AMK)
Koulutusohjelma	Tietotekniikka
Suuntautumisvaihtoehto	Ohjelmistotekniikka
Ohjaaja(t)	Lehtori Miikka Mäki-Uuro
<p>Insinööriyön aiheena on muistin parantaminen. Aihe on ajankohtainen tiedon määrän kasvassa jatkuvasti.</p> <p>Työssä esitellään muistipalatsi-niminen muististrategia ja prototyyppi ohjelmasta, jota voidaan hyödyntää muististrategiaa käyttäessä. Ohjelma luo proseduraalisesti asuntoja, joissa muistaja voi liikkua FPS-pelin tavoin. Asuntojen pohjapiirustukset ovat satunnaisesti luotuja, joten käyttäjän ei tarvitse suunnitella asuntoja.</p> <p>Työ on jaettu neljään lukuun. Ensimmäisessä luvussa kerrotaan muististrategian hyödyntämisestä, sen historiasta sekä siitä, kuinka se keksittiin ja minkälaisia versioita muististrategiasta aikojen saatossa on ollut. Seuraava luku selostaa, kuinka aivot mahdollistavat muististrategian käytön ja käy läpi tutkimuksia, jotka todistavat muististrategian hyödyllisyyttä. Kolmannessa luvussa esitellään proseduraalista sisällöntuotantoa. Viimeisessä luvussa käydään läpi prototyyppiä, joka ohjelmoitiin insinööriyön aikana ja prototyypin parannusehdotukset.</p> <p>Työn aikana saatiin luotua käytettävä prototyyppi. Prototyyppi luo kalusteettomia huoneistoja. Seuraavat tehtävät ovat kalustus-algoritmin toteuttaminen ja mahdollisuus muokata huoneiston ominaisuuksia kehitysympäristöstä. Nyt muutokset huoneistoon tehdään lähdekoodia muokkaamalla. Työ olisi voitu suorittaa taloudellisemmin suunnittelemalla työn vaiheet tarkemmin.</p>	
Avainsanat	muisti, proseduraalinen, sisällöntuotanto, proseduraalinen sisällöntuotanto

Author(s) Title	Ville-Pekka Ranta-aho Procedural Generation of a Memory Palace
Number of Pages Date	28 pages + 1 appendix 2 December 2016
Degree	Bachelor of Engineering
Degree Programme	Information Technology
Specialisation option	Software Engineering
Instructor(s)	Miikka Mäki-Uuro, Senior Lecturer
<p>The subject of this bachelor's thesis is the improvement of human memory. This subject is current, as the amount of knowledge produced seems to grow every year.</p> <p>The thesis presents a memory strategy called method of loci and a prototype of a program which can be used with the memory strategy. The prototype builds apartments procedurally where the user can move around in the same style as in FPS-games. The floor plans of the apartments are randomly generated so the user does not have to design the apartments.</p> <p>The study is divided into four parts. The first reviews the history of the strategy, how it was invented, how it is used and what types of different versions of it there have been. Next, the study explains how the brain produces the capability to use this strategy and explores some research which speaks for the effectiveness of the strategy. Thirdly the study presents the subject of procedural content generation and finally reviews the programmed prototype and offers improvement suggestions to the program.</p> <p>A working prototype, which can create empty apartments, was built. Next tasks would be to implement a furnishing algorithm and add the possibility to change apartment's properties from Unity without the need to change the source code directly. A lack of proper planning prolonged the writing of the thesis. Without a proper plan, there was a tendency to switch back and forth between tasks. This caused avoiding of seemingly hard tasks.</p>	
Keywords	memory, procedural, content generation,

Sisällys

1	Johdanto	1
2	Muistipalatsi-tekniikan historia	2
2.1	Muististrategian juuret	2
2.2	Muististrategia keskiajasta nykyaikaan	3
3	Muististrategian teoriapohja	9
3.1	Aivojen rakenne	9
3.2	Psykologiset ilmiöt	11
4	Proseduraalinen sisällöntuotanto	14
5	Virtuaalisen muistipalatsin rakentaminen	16
5.1	Kehitysympäristö	17
5.2	Algoritmi pohjapiirustuksen luomiseksi	17
5.3	Huoneiden yhdistäminen	19
5.4	Algoritmin toteuttaminen	20
5.5	Sisustuselementtien luonti	22
5.6	Huoneiden kalustaminen	24
6	Yhteenveto	25
	Lähteet	26
	Liitteet	
	Liite 1. Pohjapiirustuksen luonti -algoritmin kulku	

1 Johdanto

Tässä insinööriyössä käsitellään muistipalatsi-nimistä muististrategiaa. Työn aikana kehitettiin prototyyppi sovelluksesta, joka luo ympäristöjä muistipalatsi-strategiaa varten.

Muistipalatsi-strategia perustuu ihmisen kykyyn muistaa paremmin konkreettisia asioita. Muistipalatsi itsessään on rakennus, joka henkilöllä on selkeästi muistissa. Strategiassa henkilö sijoittaa mielessään muistipalatsiin haluamansa asiat.

Muististrategiasta käydään läpi sen historia, kuinka sitä käytetään, ja teorioita, minkä takia se toimii. Näiden lisäksi esittelen teknologioita, joita yhdistelemällä voidaan luoda automaattisesti suuri määrä erilaisia huoneistoja. Näitä teknologioita kutsutaan yleisesti nimellä proseduraalinen sisällöntuotanto.

Työ on jaettu neljään lukuun. Ensimmäinen luku kertoo muistipalatsi-strategian historiasta. Luku kertoo muististrategian keksimisestä sekä siitä, mihin eri tarkoituksiin sitä on käytetty ja miten se on säilynyt ihmisten tiedossa nykypäivään saakka.

Toinen luku on teoriapainotteinen. Se kertoo yleisesti ihmisen aivoista ja siitä, kuinka muisti toimii. Näiden lisäksi käydään läpi, minkä takia muistipalatsi-strategia toimii asioiden mieleenpainamisessa.

Kolmannessa ja neljännessä luvussa käydään läpi virtuaalisen muistipalatsin luomisessa käytettäviä tutkimusartikkeleita ja kerrotaan prototyypin ohjelmoimisesta. Ohjelman tarkoituksena on luoda huoneistoja erilaisilla pohjapiirustuksilla. Luvussa esitellään kehitetty sovellus, joka luo huoneistoja satunnaisilla pohjapiirustuksilla. Tämän lisäksi käydään läpi menetelmät sisustuselementtien luontiin ja niiden sijoittamiseen.

2 Muistipalatsi-tekniikan historia

2.1 Muististrategian juuret

Muistipalatsi-tekniikan löytäjäksi on kolmessa eri aineistossa mainittu kreikkalainen runoilija Simonides Keoslainen (noin 556 - 468 eaa.) (Thomas 2014; Yates 2014; Rivers 2010). Tarinan mukaan Simonides oli juhlassa, kun viestintuojat pyysivät häntä tulemaan ulos kuulemaan viestin. Ulkona ollessaan juhlasali romahti ja vieraat kuolivat. Ruumiit olivat tunnistamattomia, mutta Simonides kykeni visualisoimalla muistamaan, missä kukin istui. Tämän avulla ruumiit saatiin tunnistettua. Tämän kokemuksen perusteella Simonides päätteli, että muistia parantaakseen tulee henkilön valita paikkoja, muodostaa mielikuvia muistettavista asioista ja sijoittaa asiat paikkoihin.

Muistitekniikka oli yksi osa puhetaidon, retoriikan, opiskelua. Muistin kehittäminen oli antiikin kreikan aikaan kriittinen osa retoriikkaa, koska paperia tai papyrusta ei ollut välttämättä saatavilla. Kaikki tätä muistitekniikkaa kuvaavat teokset pohjautuvat tuntemattoman kirjailijan teokseen *Ad Herennium*. Teoksessa hän puhuu retoriikan opiskelijoille ja kuvailee sääntöjä, joiden mukaan muistipalatsi tulee luoda ja miten sijoittaa muistettavat asiat.

Kun muistipalatsi on rakennettu, sitä voidaan käyttää vuorotellen eri asioiden muistamiseen. Lisäksi hän kuvailee, että huoneiden tulisi erota keskenään sekaantumisen välttämiseksi. Jotta sijainnit olisi helpompi muistaa, opettaja kehottaa sijoittamaan viiden paikan välein sijainnin järjestysnumeroa esittävän kuvan. Tämä palauttaa mieleen muistipaikan järjestysnumeron jolla henkilö on, tai hän voi mennä muutaman paikan eteen- tai taaksepäin selvittääkseen sijaintinsa. Sopiva etäisyys muistipaikkojen välillä on opettajan mukaan noin 10 metriä. Muistipaikan tulisi myös olla sopivasti valaistua, ei liian kirkas eikä liian pimeä.

Seuraavaksi teoksessa kuvataan sääntöjä, joiden perusteella muistipaikkoihin sijoitettavia muistikuvia luodaan. On kahdenlaisia kuvia: toiset kuvaavat puheenaiheita ja toiset yksittäisiä sanoja. Opettajan mukaan kuvien tulisi olla mahdollisimman tunteita herättäviä ja arjesta poikkeavia. Emmehän itsekään muista kaikkea arkisia tapahtumia, koska niissä ei tapahdu mitään poikkeavaa. Opettaja kannustaa tekemään kuvista erittäin kauniita tai rumia ja lisäämään esimerkiksi verta tai likaa mielikuviiin.

Kirjassaan *De Inventione* ("Argumenttien rakenteesta") Ciceron määritelmä hyveestä on "mielen tapa, joka on harmoniassa järjen ja luonnonlakien kanssa". Hyve jakaantuu neljään osaan, joita ovat harkitsevuus, oikeudentaju, luonteenlujuus ja itsehillintä. Näistä jokainen vielä jakaantuu osa-alueisiin. Harkitsevuuden Cicero määrittelee "ymmärrykseksi mikä on hyvää, mikä pahaa, ja mikä ei ole kumpaakaan. Sen osia ovat muisti, äly ja kaukokatseisuus.

Aristoteles oli tietoinen muistipalatsi-järjestelmästä. Hän mainitsee teoksessaan *Topica* ("Topiikka"), kuinka argumentoijalle palaa mieleen tietyn paikan kuullessaan sen paikan sisältämä tieto. Aristoteles mainitsi *De memoria et reminiscentia*-teoksessaan ensimmäisenä aivojen assosiatiivisuuden eli tavan, jolla mieli saattaa kyetä palauttamaan halutun asia tietosuuteen.

Aristoteleen tietoteoria perustui aisteihimme. Mieli saa aistien kautta tietoa. Tätä tietoa mielikuvitus käsittelee, josta myöhemmin tulee osa meidän ymmärrystämme ja tietomme maailmasta.

Platonin teoria tiedosta oli, että ennen syntymää meillä on synnynnäistä tietoa korkeammasta todellisuudesta, joka ilmenee muun muassa tasa-arvoisuuden käsitteenä. Giulio Camillo yritti luoda Platonin muistiteorian mukaan Muistiteatteria, joka sisältäisi kaiken tiedon luonnosta ja ihmisestä.

Aristoteles ja Platon vaikuttivat molemmat myöhempään muistipalatsin käyttöön ja opiskelamiseen. Aristoteleelta ottivat vaikutteita enemmän keskiajan ja skolastiikan ajan muistipalatsin käyttäjät, ja Platon vaikutti renessanssin aikakauden muistitaidon harrastajiin.

2.2 Muististrategia keskiajasta nykyaikaan

Martianus Capella kirjoitti kirjan *De nuptiis Philologiae et Mercurii*, joka säilytti antiikin vapaat tieteet keskiajalle. Vapaat taidot jaetaan seitsemään osaan, kielioppi, retoriikka, dialektiikka, aritmetiikka, geometria, musiikki ja astronomia. Mary Carruthers päätyi samaan tulokseen Yatesin kanssa, eli kummankaan, Simonideksen tai Quintilianuksen muistitaidoista ei löytynyt laajaa keskustelua aikavälillä 400-1100 jaa (Rivers 2010: 19).

Capellan kirja ei kuitenkaan ollut pääsyy keinotekoisien muistin säilyvyyteen, vaan muistipalatsi-järjestelmä oli edelleen käytössä antiikin sivistyneistön piireissä. Keskiajalla Ad Herennium liitettiin Ciceroon. Ciceron teosta De Inventione nimitettiin "First Rhetoric":ksi ja Ad Herenniumia "Second Rhetoric":ksi. De Inventione keskittyi etiikkaan ja hyveisiin puheissa, kun taas Ad Herennium kuvasi, kuinka muistettavat asiat painetaan mieleen.

Albertus Magnus ja Tuomas Akvinolainen tutustuivat muistin harjoittamiseen Ciceroon liitetyn Ad Herenniumin ja De Inventionen kautta. Näihin aikoihin muistin harjoittaminen siirtyi muistin hyödyntämisestä puheissa ja väittelyissä hyveiden muistamiseen ja harjoittamiseen.

Albertus Magnus yhdisti ensimmäisenä Aristoteleen ajatukset muistista ja Ad Herennium teoksen opit. Keskiajalla Albertus Magnuksen lisäksi Tuomas Akvinolainen kirjoitti muistin harjoittamisesta osana hyveiden kehittämistä. Kirjoissaan hän muun muassa pohti ja esitti argumentteja muistin parantamisen puolesta hyveiden harjoittamista varten. Jotkut keskiajan henkilöt olivat nähtävästi sitä mieltä, että muisti oli synnynnäinen ominaisuus, jota ei tarvitse parantaa.

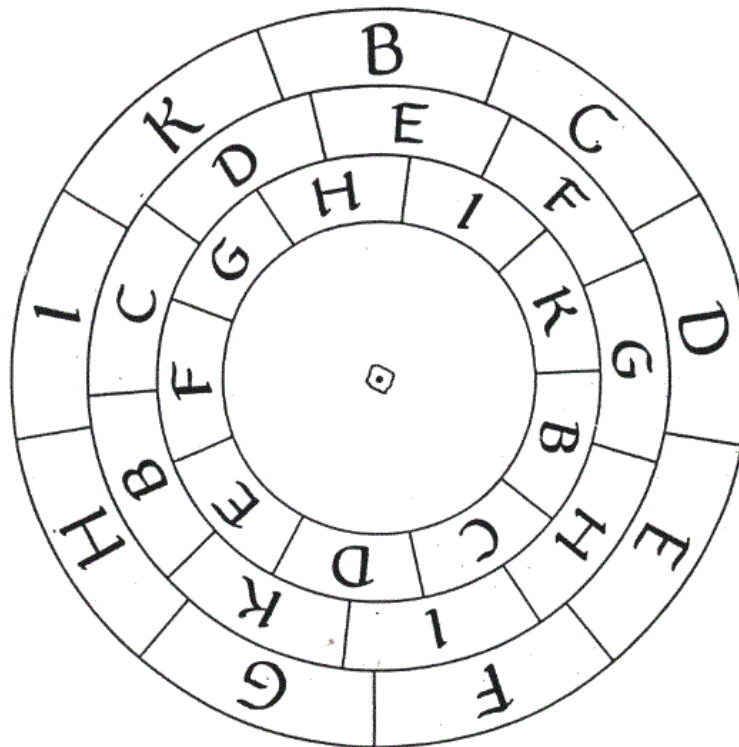
Kirjapainon yleistyessä materiaalin määrä kasvoi suuresti. Yatesin mukaan renessanssin humanistit suhtautuivat nihkeästi muistitaitoihin niiden tarpeellisuudesta huolimatta. Tämän mielipiteen humanistit saivat Quintilianuksen kirjoituksista, jonka töitä humanistit lukivat. Yksi näistä humanisteista johon Quintilianus saattoi vaikuttaa, oli Erasmus Rotterdamilainen. Hänen mielestään muistipalatsia saattoi hyödyntää, mutta oppimisen kannalta tärkeimpiä asioita ovat opiskelu, järjestys ja huolellisuus. Quintilianus suhtautui ylimielisesti ja kriittisesti muistitaitoihin. (Yates 2014: 132-133.)

1500-luvulla muistitaidot siirtyivät osittain marginaaliseksi kirjapainon ja humanistien nihkeän suhtautumisen takia. Muistitaidon perinteitä rupesivat seuraavaksi viemään eteenpäin uusplatoninen liike. Uusplatonisen filosofian, hermetismin, ydin vaikutti siihen, miten muistitaidot muuttuivat.

Ramon Lullin muistitaito alkoi keskiajalla ja jatkui renessanssista eteenpäin. Lull levitti taitoaan samaan aikaan, kun Albertus Magnus ja Tuomas Akvinolainen suosittelivat klassisen taidon keskiaikaista versiota. Lull sai idean muistitaitoonsa nähtyään näyn Jumalan ominaisuuksista.

Lullin muistitaito perustuu Jumalan nimiin Bonitas, Magnitudo, Eternitas, Potestas, Sapientia, Voluntas, Virtus, Veritas ja Gloria. Nämä muodostavat taidon perustan. Konsepteja vastaavat kirjaimet BCDEFGHIK. Yleiset Jumalan nimet ovat samat kristityillä, juutalaisilla ja muslimeilla. Taidon kosmologinen rakenne perustuu tieteellisiin, hyväksytyihin konsepteihin.

Kuva 1 on lullismissa käytetystä muistamismetodista. Kuvan kehiä pyöritetään mielessä ja luodaan erilaisia kirjainyhdistelmiä.



Kuva 1. Ramon Lullin muistamiseen tarkoitettu pyörä (Anthony Bonner 1997).

Lull kirjoitti yhden teoksen muistista nimeltään "Liber ad memoriam confirmandam". Teoksessaan hän jakaa muistin Ad Herennium-teoksen tapaan kahteen osaan: synnynnäiseen ja keinotekoiseen muistiin. Vaikka muistityyppien jako viittaa hänen tietäneen aiempia muistista kertovia teoksia, hän sanoo ainoaksi muistamisen säännöksi säännöllisen asian pohtimisen ja toistamisen.

Giordano Bruno liittyi dominikaaneihin, eli saarnaveljiin 1563 ja opiskeli dominikaanien luostarissa Napolissa. Hän kuitenkin karkasi luostarista ja keskittyi kehittämään omaan muistimenetelmäänsä. Menetelmässään hän yhdisteli taikuutta ja dominikaanien muistitaitoja.

Bruno yhdisteli okkultismia molempiin, sekä perinteiseen muistitaitoon että lullismiin. Perinteisen muistitaidon kuville tuli maagisia ominaisuuksia ja renkailla pyrittiin kutsumaan henkiä. Renkaiden kuvia hän otti egyptiläisistä tähtiin perustuvista taruista. Kuvat perustuivat Heinrich Cornelius Agrippan kirjaan *De occulta philosophia*. Agrippan kirjan mukaan Brunon käyttämät kuvat olivat tapoja toimia taivaallisen maailman kanssa.

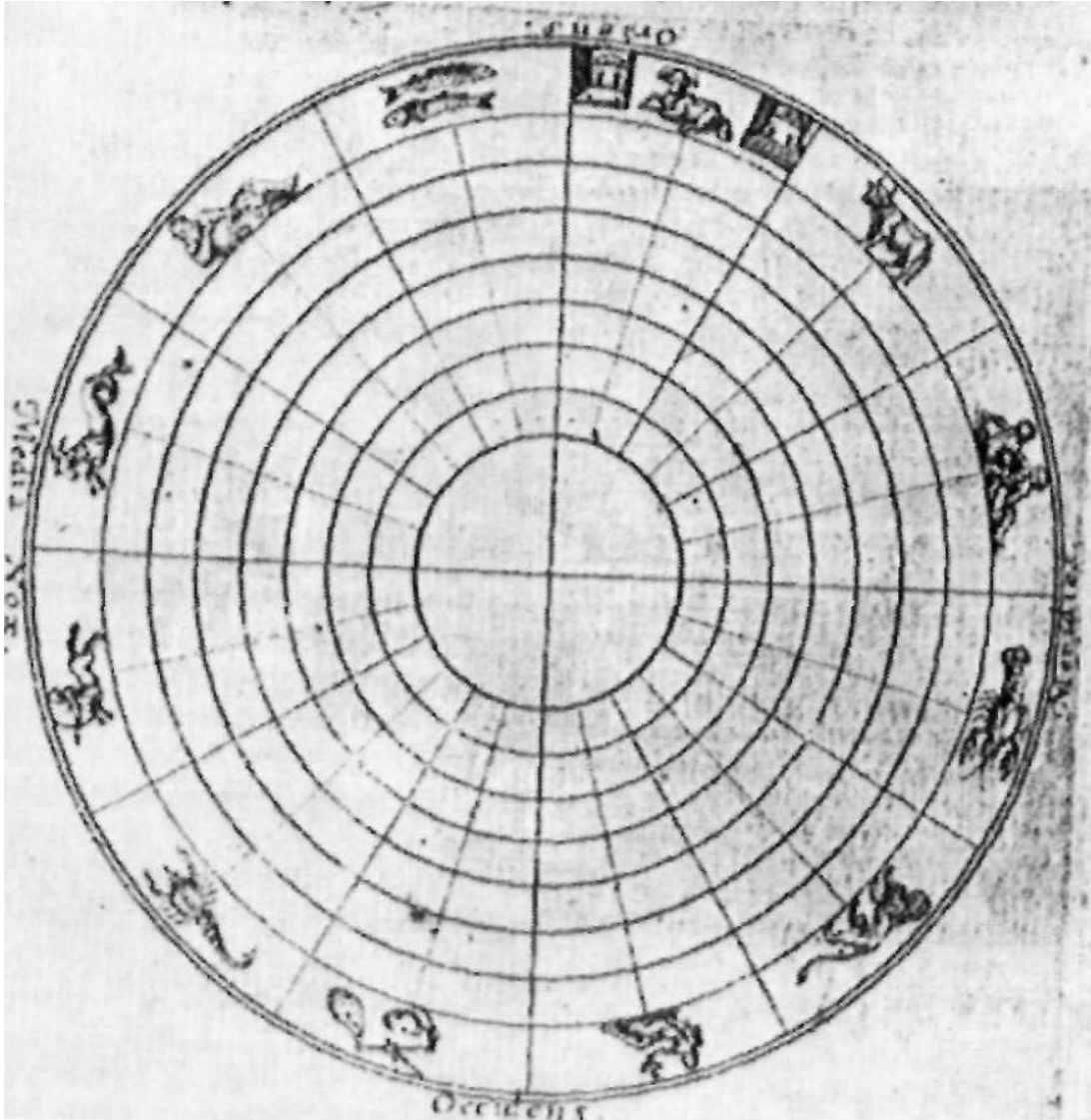
1500-luvulla okkultistinen muistitaito kasvatti suosiotaan perinteisen menetelmän suosion vähentyessä. Esimerkiksi humanistiopettaja Melanchton jätti muistitaidon pois retoriikan opetuksesta. Yates ehdottaa perinteisen muistitaidon suosion heikentymisen syyksi, että se nähtiin vanhahtavana sen tullessa keskiajalta, jossa sitä käyttivät pääasiassa saarnaajat.

Petrus Ramus, joka Yatesin mukaan uudisti opiskelumenetelmiä, pyrki yksinkertaistamaan asioiden mieleen painamista. Tämä tapahtui järjestämällä opiskellun aineen asiat johonkin järjestykseen.

Myöhäisrenessanssin aikaan sekä Ramus että Bruno uudistivat asioita muistitaidon avulla. Ramus uudisti koulutusta dialektisella järjestelyllä ja Bruno hermetismiä mielikuvilla ja mielikuvituksella.

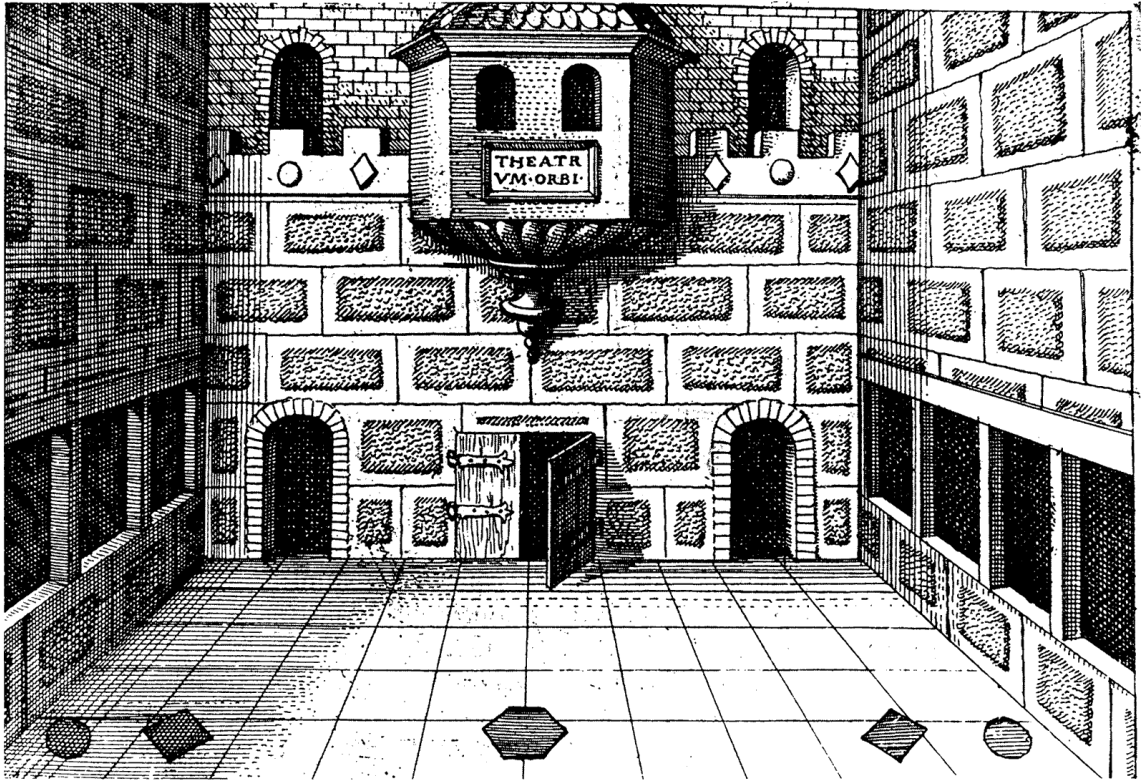
Robert Fluddin järjestelmässä on kaksi osaa, pyöreä ja neliö. Pyöreä osa käyttää taikaja talismaanikuvia, tähtiä, patsaita jumalattarista ja jumalista sekä kuvia hyveistä ja pahaista. Neliö osa käyttää konkreettisia asioita, ihmisiä, eläimiä ja esineitä. Kuvien kohteet tekevät jotakin. Nämä kaksi osaa Fludd yhdistää sijoittamalla neliön osia pyöreisiin.

Fludd antaa ennalta luodut esimerkit muistipalatsin paikoille. Kuvassa 2 on Fluddin pyöreän osan muistipaikkaesimerkki.



Kuva 2. Fluddin horoskooppi rengas (Yates 2014).

Teatterista on kaksi eri versiota, itäinen ja läntinen. Itäinen on kirkas ja valoisa, kun taas läntinen on synkkä ja pimeä. Yates epäilee, että nämä kaksi huonetta sijoitetaan jokaiseen horoskooppiin. Tällä tavalla saataisiin 24 muistiteatteria, joista jokaiseen voidaan sijoittaa usea muistettava asia. Kuvassa 3 on Fluddin muistiteatteri.



Kuva 3. Fluddin muistiteatteri (Yates2014).

Teatterissa on viisi ovea. Jokainen ovi on väritään yksilöllinen. Näitä ovia käytetään muistipaikkoina. Ovien edessä on eri mallisia pylväitä, joiden värit vastaavat niiden edessä olevia ovia. Näitä pylväitä voidaan myös käyttää muistipaikkoina.

Fluddin järjestelmään saattoi vaikuttaa suoraan John Willisin vuonna 1618 julkaistu *Mnemonic; sive Ars reminiscendi*. Willis käyttää samankaltaisia teattereita muistamiseen ja kehottaa myös käyttämään värejä erottamaan teatterit toisistaan.

Hermetismin suosion heikentyessä 1600-luvulla nousi uusi ryhmä hyödyntämään muistitaitoja. Muistitaitoja ryhdyttiin käyttämään uuden tiedon löytämiseksi. Francis Bacon oli tietoinen muistitaidoista ja kirjoitti niistä kirjassaan "Advancement of Learning". Baconin mukaan muistipalatsi-strategiaa voitaisiin parantaa ja sitä tulisi käyttää hyödyllisiin tarkoituksiin.

Yates kertoo Gottfried Leibnizin perehtyneen lullismiin ja hyödynsi Lullin ideoita matematiikkaan. Leibniz kehitti universaalia kalkyyliä. Tällä metodilla oltaisiin voitu laskemaan

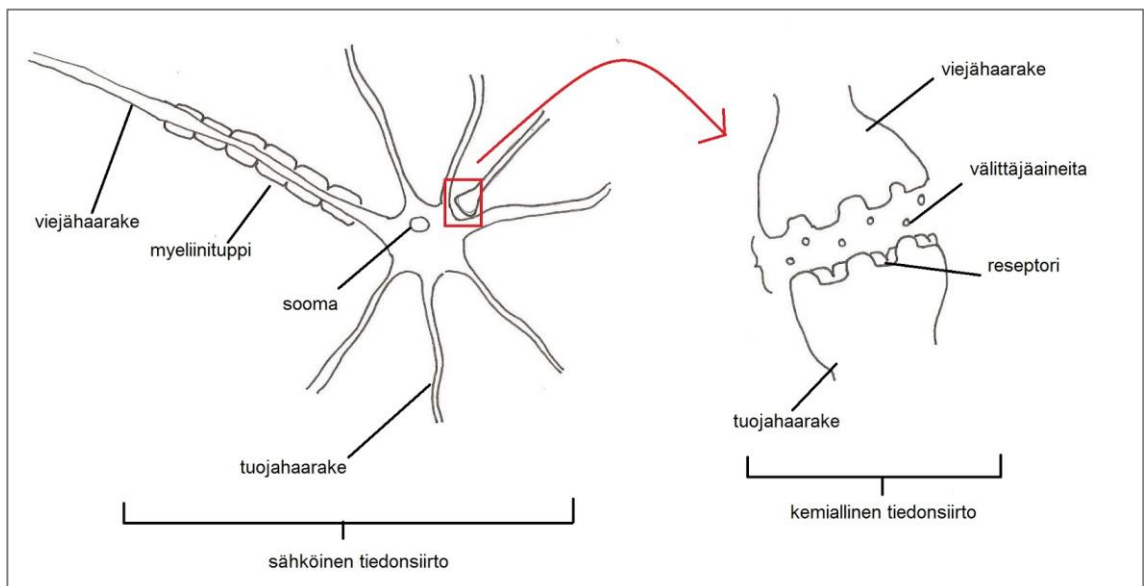
ajatusten oikeellisuus. Yates lopettaa aiheen läpikäymisen Leibniziin, mutta mainitsee, että teoksia muistitaidon perinteisestä versiosta julkaistiin myöhempinäkin vuosisatoina.

Nykyajan teoksista luin muun muassa Derren Brownin suomennetun kirjan ”Miten mieltä hallitaan”, jossa hän käy läpi mentalistien eri temppuja. Muistipalatsin hän opettaa pitkälti vanhojen ja perinteisten oppaiden tapaan. Poikkeuksena on hänen mainintansa, ettei muistipaikkojen etäisyydellä ole niinkään merkitystä vaan sillä, että paikat erottuvat selkeästi toisistaan (Brown 2013: 99).

3 Muististrategian teoriapohja

3.1 Aivojen rakenne

Aivot koostuvat neuroneista ja näiden tukisoluista. Neuroneilla on solurunko, aksoni ja dendriittejä. Dendriitit ovat kuin johtoja, jotka tuovat viestejä solurunkoon. Aksonit lähettävät sähkövirtaa eteenpäin solurungosta seuraavalle neuronille.



Kuva 4. Hermosolu (Johanna Mattila <http://creativecommons.org/licenses/by-sa/4.0/>).

Aksonit ja dendriitit on eristetty rasvakerroksella, jota nimitetään myeliiniksi. Tämä kerros nopeuttaa aktiopotentialin kulkua. Aktiopotentiali on sähköinen viesti, joka kulkee dendriiteissä ja aksoneissa.

Aksonin ja dendriitin kohtausta paikkaa nimitetään synapsiksi. Synapsia pitkin viestit etenee ainoastaan yhteen suuntaan, aksonista seuraavan hermosolun dendriitille. Kun aksonin päähän saapuu aktiopotentiali, se vapauttaa molekyylejä, jotka kulkevat synapsiraon yli dendriitin osille. Näitä molekyylejä kutsutaan hermojen välittäjäaineiksi, joita ovat esimerkiksi serotoniini ja dopamiini. Se että yhdestä hermosolun dendriitistä tulee aktiopotentiali solurungolle ei välttämättä riitä solun aktivoimiseksi. Saatetaan tarvita usean dendriitin samanaikainen viesti, jotta aktiopotentiali etenee solurungosta eteenpäin. Pääasiallinen viestintäsarja on siis sähkö-kemia-sähkö, mutta myös pelkästään sähköisesti toimivia viestintäsarjoja on (LeDoux 2003: 61).

Neuroneista eli hermosoluista rakentuu piirejä. Näiden piirien hermosoluja kutsutaan välineuroneiksi. Yksi esimerkki on verkkokalvon piiri, joka välittää verkkokalvoille tulevan informaation eteenpäin. Sitten on projektiosoluja, jotka pyrkivät käynnistämään hierarkian seuraavan piirin. (LeDoux 2003: 61.)

Teoriaosuus perustuu Jeff Hawkinsin muisti-ennustus -malliin. Hawkinsin muisti-ennustus -teorian mukaan aivot luovat jatkuvasti uusia malleja ja muokkaavat vanhoja tapahtumien perusteella. Samaan aikaan, kun aivot päättävät mitä tehdä, se ennustaa, mitä tulee tapahtumaan. Mikäli aivot ennustavat oikein, mitään ei tarvitse muuttaa, mutta jos ennustus oli väärä, muutetaan mallia. Kerrataan lyhyesti antiikin kreikan ohjeet muistamiseen. Painetaan mieleen rakennus, jota halutaan käyttää muistamiseen. Huoneiden tulee olla hyvin valaistut ja toisistaan eroavat. Kuljetaan talon läpi aina samassa järjestyksessä. Tehdään muistettavista mielikuvista mahdollisimman arjesta poikkeavia ja tunteita herättäviä.

Jeff Hawkins kirjassaan *Älykkyys* esittää muistin mahdollista toimintatapaa. Ehdotuksen mukaan kaikki kokemukset ovat sarja neuronien laukeamisia. Kun sama asia koetaan, samat neuronit laukeavat samassa järjestyksessä. Hiljalleen aivot oppivat oletamaan, mitkä neuronit laukeavat ja milloin. Tietyn asian muistamiseen aivot eivät tarvitse kuin osan muistettavasta asiasta, niin aivot osaavat täydentää ja täten muistaa haluttu asia. (Hawkins 2005.)

3.2 Psykologiset ilmiöt

Psykologiset ilmiöt, jotka tukevat muistipalatsi-strategian käyttöä ovat, esimerkiksi “picture superiority effect”, jonka mukaan ihmiset muistavat paremmin kuvia kuin sanoja, Von Restorffin ilmiö ja assosiativisuus.

Picture superiority on ilmiö, jonka mukaan ihmiset tunnistavat ja muistavat paremmin näkemänsä kuvat kuin sanat. Higbee esittelee julkaisussaan “Recent research on visual mnemonics” (1979) ilmiötä tukevia tutkimuksia. Tätä selittäviä teorioita ovat muun muassa Päiviön kaksoiskoodusteoria. Kyseisen teorian mukaan aivoissa on kaksi järjestelmää, jotka koodaavat aistimuksia, eli tallentavat muistijäljet, verbaalinen ja kuvallinen. Nämä järjestelmät ovat yhteydessä toisiinsa, joten molemmat voivat käynnistää toisen muistijälkiä. Päiviö ehdottaa, että kuvat käynnistävät muistikuvia molemmissa järjestelmissä helpommin kuin sanat, koska kuva-järjestelmä prosessoi muistijälkiä rinnakkain, kun verbaalinen järjestelmä prosessoi sarjassa. (Paivio 2006.)

Monien muiden mielestä ilmiön saa aikaan kuvien laajempi semanttinen prosessointi. Tämä tarkoittaa, että valokuva herättää katsojassa enemmän assosiaatioita kuin vastaava sana. (Mintzer ym. 1999.)

Von Restorffin ilmiötä kutsutaan myös “isolation effect”:ksi. Gilles O. Einstein ja Mark A. McDaniel esittelivät katsauksen aiheesta (McDaniel ym. 1987: 78). He jakoivat tutkimukset kahteen osaan. Toisessa testattiin muistamista alle viisi minuuttia mieleen painamisesta ja toisessa ryhmässä aikaväli vaihteli viidestä minuutista kahteen viikkoon. Tutkimuksissa koehenkilöt jaettiin ryhmiin ja osalle annettiin lista normaaleja lauseita, joillekin lista kummallisia lauseita (esimerkiksi KOIRA ajoi PYÖRÄLLÄ pitkin KATUA) ja joidenkin listassa oli sekä normaaleja että kummallisia lauseita. Näistä lauseista heidän tuli luoda mielikuvat tai pelkästään arvioimaan lauseen kummallisuus ilman mielikuvan luontia. Tauon jälkeen heidän tuli palauttaa mieleen lauseiden substantiivit.

Lyhyen aikavälin tutkimuksissa havaittiin kolmen asian vaikuttavan tuloksiin. Mikäli listassa oli normaaleja ja kummallisia lauseita sekaisin, ne muistettiin paremmin kuin jos täytyi muistaa pelkästään yhden tyyppisiä lauseita. Toinen asia oli, palauttaako koehenkilö asian itse mieleen vai annetaanko hänelle vinkkejä. Viimeinen vaikuttava asia oli, miten koehenkilö käsittelee lausetta, luoko hän siitä mielikuvan vai arvioiko hän lauseen kummallisuuden.

Pitkän tauon (tutkimuksissa tauko vaihteli 5 minuutista kahteen viikkoon) tutkimuksissa oli viitteitä siitä, että kummallisten lauseiden substantiivit muistettiin paremmin kuin normaalien lauseiden. Näistä oli kuitenkin eri tutkimuksissa ristiriitaisia tuloksia.

Tutkijoiden johtopäätöksen perusteella ei vielä tiedetä tarkalleen, kuinka paljon hyötyä kummallisten mielikuvien luomisesta on ja minkälaisissa konteksteissa niiden käyttäminen toimii. Järjestelmällisiä tutkimuksia ilmiöstä tarvitaan enemmän.

J. B. Worthen ja R. R. Hunt käyvät kirjan "Learning and Memory: A Comprehensive Reference" luvussa 2.10 läpi prosesseja, joita ihminen tekee muistiteknikoita hyödyntäessään. Prosesseja on listattu kolme erilaista: organisointi, kehittäminen ja mielikuvat.

Organisoinnilla tarkoitetaan muistettavan tiedon järjestämistä helpommin muistettavaan muotoon. Muistipalatsi-strategiassa organisointi tapahtuu sijoittamalla muistettavat mielikuvat tiettyssä järjestyksessä mieleen painamalla reitille tai rakennukseen. (Jenkins ym. 1952; Bowler ym. 1969; Broadbent ym. 1978.)

He ehdottavat, että tiedon organisointi vähentää vaatimuksia, joita tiedon säilytys aiheuttaa. Tämä tapahtuu järjestämällä materiaali muotoon, joka muistuttaa mahdollisimman paljon pitkäaikaisen muistin rakennetta.

Kehittäminen liittyy "levels of processing" -ilmiöön. Kehittämisellä tarkoitetaan aiemmin sisäistetyn tiedon sovittamista muistettavan asian kanssa yhteen. (Craik ym. 1975; Rogers ym. 1977; Nelson, 1979) Tulvingin ja Craikin tutkimuksessa koehenkilöitä kehoitettiin pohtimaan annettuja sanoja eri "syvyyksillä". Prosessoinnin syvyyksiä oli kolme:

Pinnallinen, jossa pyydettiin analysoimaan sanan ulkonäköä (esimerkiksi onko teksti kirjoitettu isoilla kirjaimilla). Keskitaso, jossa kysyttiin rimmaako näytetty sana jonkin toisen kanssa. Syvä: semanttinen analyysi, jossa pyydettiin joko luokittelemaan sana, esim. "onko sana eläimen nimi?" tai sanomaan sopiiko näytetty sana annettuun lauseeseen. Kokeissa todettiin sanan syvemmän prosessoinnin parantavan sanan muistamista.

Myös asioiden liittäminen, joilla ei ole merkittävää yhteyttä muistettavaan asiaan, on havaittu toimivaksi muististrategiaksi. (Slamecka ym. 1978; Kolers 1979; Pressley ym. 1987)

Worthenin ja Huntin mukaan mielikuvat auttavat parantamaan muistia erityisesti silloin, kun mielikuvissa olevat esineet vaikuttavat toisiinsa. Heidän mukaansa syynä on mielikuvien mahdollisuus organisoida ja kehitellä muistettavaa tietoa entistä enemmän. Tämä edesauttaa asian enkoodausta ja säilöntää, jolloin asia on helpompi palauttaa mieleen.

Artikkelissaan "Routes to remembering: the brains behind superior memory" (Maguire ym. 2002) Maguire ym. tutkivat, mitä eroja huippumuistajien (henkilöt jotka olivat voittaneet muistamisen maailmanmestaruuskisoja) ja verrokkiryhmän välillä on. He vertasivat henkilöiden älykkyyttä, aivojen rakenteellisia eroja ja viimeisenä toiminnallisia eroja. Ainoa osa-alue, josta tutkijat löysivät eroja, oli aivojen toiminta.

Muistamista tutkijat testasivat kolmella tavalla. Koehenkilöiden tuli yrittää muistaa kolmi-numeroisia lukuja, kasvoja ja erilaisia lumihuutaleita. Jokaisen asian muistaminen testattiin erikseen, eli numerot omana kokeenaan, kasvat omanaan jne.

Huippumuistajilla aktivoituivat pikkuaivot voimakkaammin kuin verrokkiryhmällä. Ainoastaan huippumuistajilla aktivoitui seuraavat kolme aivojen osaa, vasen sisempi ylempi päälaen aivopoimu (left medial superior parietal gyrus), aivokurkiaisien takana oleva aivokuoren osa (retrosplenial cortex) ja oikea taaempi hippokampuksen osa (right posterior hippocampus).

Vasen sisempi ylempi päälaen aivopoimu on ainakin yksi aivojen osa, mikä on vastuussa työmuistissa olevan tiedon manipuloinnista (Koenigs ym. 2009). Tätä kykyä tarvitaan, kun saatua tietoa halutaan muokata paremmin muistettavaan muotoon. Tätä prosessia kutsutaan kehittelyksi.

Vannym. (2009) kävivät läpi retrosplenial cortexiin liittyviä tutkimuksia. Retrosplenial cortexilla on rooli ainakin navigoinnissa ja spatiaalisessa muistissa. Samoin oppiminen verbaalisen ja visuaalisen tiedon perusteella on hankaloitunut. Retrosplenial cortexin vaurioitumisessa on ilmennyt myös vaikeutta oppia navigoimaan uusissa paikoissa. Hippokampuksen vaurioissa navigointi on myös vaikeutunut, mutta he osaavat kulkea tutuissa ympäristöissä ja kykenevät säilyttämään suuntavaiston. Yksi yleinen teoria on, että retrosplenial cortex kääntää allosentrisen tiedon egosentriseksi ja toisin päin. Allosentrisen tarkoittaa, miten ympäristön esineet ovat suhteessa toisiinsa, ja egosentrisen, minkälaisessa suhteessa ne ovat havaitsijaan.

Nadel ym. (2013) analysoivat fmri-kuvauksen tuloksia. He havaitsivat hippokampuksen takaosan aktivoituvan koehenkilöiden muistellessa tarkkaa spatiaalista tietoa, kun taas hippokampuksen etuosa aktivoitui hakiessa kontekstiin liittyvää informaatiota (Chua ym. 2007). Nämä tulokset tukevat Maguiren ym. tulosta, jossa huippumuistajien käyttäessä muistipalatsi-strategiaa muun muassa heidän oikea taaempi hippokampuksen osa aktivoitui.

4 Proseduraalinen sisällöntuotanto

Proseduraalinen sisällöntuotanto tarkoittaa, että on jokin algoritmi, joka tuottaa itsenäisesti tai vähäisellä ihmisen panostuksella sisältöä. Togelius ym. (2011b) luokittelevat lähes kaiken muun paitsi pelitekoälyn ja pelimoottorin pelin sisällöksi. Tässä työssä proseduraalista sisällön tuottamista käytetään asuntojen luomisessa. Kyseistä tekniikkaa on käytetty useissa videopeleissä sen taloudellisuuden takia; algoritmi kykenee itsenäisesti luomaan useita vaihtoehtoja säästäten työtunteja. Vaikka tässä työssä proseduraalista sisällöntuotantoa on käytetty ainoastaan asunnon pohjapiirustuksen luomiseen, sitä voidaan hyödyntää useisiin muihin pelin osa-alueisiin. Näitä osa-alueita ovat muun muassa tekstuurit, kasvit, äänet, kartat, maasto ja tieverkostot (Hendriks ym. 2013).

Proseduraalinen sisällöntuotanto madaltaa myös kynnystä pelien tekemiseen. Pelin kehittäjä voi keskittyä niihin osa-alueisiin, joista on kiinnostunut, ja antaa algoritmien luoda muut osa-alueet.

Togelius ym. kirjassaan *Procedural Content Generation in Games* käyvät läpi proseduraalisen sisällöntuoton ratkaisun haluttuja ominaisuuksia. Näitä kaikkia ominaisuuksia ei välttämättä ole mahdollista saada kerralla, vaan kehittäjän täytyy tehdä kompromisseja.

Usein halutut ominaisuudet ovat:

- Nopeus: Mikäli algoritmi luo sisällön kesken pelin, täytyy algoritmin olla erittäin nopea. Pelin suunnitteluvaiheessa algoritmin ajonopeudella ei ole yhtä suuria rajoitteita.
- Luotettavuus: Algoritmin tulee luoda tavoitteiden mukaista sisältöä. Työn ohjelman tapauksessa algoritmin tulee luoda huoneistoja, joiden kaikkiin huoneisiin pääsee.
- Hallittavuus: Hallittavuus kertoo, miten paljon pelaaja tai suunnittelija voi vaikuttaa algoritmin toimintaan.
- Ilmaisuvoimaisuus: Ilmaisuvoimaisuus kertoo, kuinka laajan kirjon vaihtoehtoja algoritmi luo. Esimerkkinä tuolin luominen. Muuttaako algoritmi ainoastaan tuolin väriä vai vaihtaako se jalkojen määrää ja muotoa, selkänöjan tyyliä tai vaikka tuolin materiaalia?
- Luovuus ja uskottavuus: Edellisen kohdan esimerkkiä käyttäen, luoko algoritmi tuoleja, joilla voisi istua.

Koska on useita eri käyttötarkoituksia proseduraaliselle sisällöntuotannolle, on katsottu järkeväksi pyrkiä luokittelemaan sisällöntuoton algoritmeja.

Togelius ym. luokittelevat pcg (procedural content generation) -algoritmeja seuraavien ulottuvuuksien mukaan.

Ajonaikainen vs. kehityksen aikainen. Ajonaikainen tarkoittaa, kun algoritmi esimerkiksi luo pelitasoa sitä mukaa, kun pelaaja liikkuu. Toisessa ääripäässä ovat algoritmit, joita käytetään pelin kehitysvaiheessa. Tällaisia voivat olla esimerkiksi algoritmit, jotka luovat kokonaisia kenttiä ja täten vaativat paljon ajoaikaa.

Luotavan sisällön tarpeellisuus. Algoritmia luotaessa on hyvä miettiä, onko algoritmin luoma sisältö tarpeellista vai vapaaehtoista. Pelihahmon tai maaston tietty ulkonäkö ei esimerkiksi ole välttämätöntä kentän läpäisyyn. Sen sijaan, jos vaikka rallipelissä luodaan proseduraalisesti rata, tulee sen lähtöruudusta olla mahdollista päästä maaliin.

Hallinnan aste ja ulottuvuudet kertovat, kuinka paljon pcg-algoritmin tuottamaan sisältöön voi vaikuttaa esimerkiksi parametrein. Esimerkiksi talonrakennus-algoritmillemme voisi antaa ainoastaan satunnaisluvun, jonka perusteella talo luodaan. Toisaalta algoritmillemme voitaisiin antaa vektori, jossa on useita parametreja: huoneiden lukumäärä, kerrosten määrä, onko huoneet kalustettu tai mitkä huoneet ovat vierekkäin.

Geneerinen sisällöntuotanto tarkoittaa tapaa, jossa algoritmi ei ota pelaajan käyttäytymistä huomioon. Geneerisen sisällöntuotannon vastakohtana on sopeutuva sisällöntuotanto, joka analysoi pelaajan käyttäytymistä ja luo analyysin perusteella erilaista sisältöä.

Stokastinen tarkoittaa satunnaista, eli stokastisen algoritmin tuloksia ei pysty ennustamaan. Deterministinen algoritmi on stokastisen vastakohta, se tuottaa aina samalla siemenellä saman lopputuloksen. Determinististä algoritmia voidaan ajatella tiedon kompressoimisena. Haluttua sisältöä ei rakenneta valmiiksi, vaan annetaan vain rakennusohjeet.

Tuottaako algoritmi sisällön automaattisesti vai vaikuttaako esimerkiksi pelaaja tai suunnittelija algoritmiin? Automaattinen tuotanto tarkoittaa, että algoritmillemme annetaan ainoastaan parametreja tai satunnaisluku, jonka perusteella se luo sisällön. Yhteisluonti on algoritmin ja sisällön suunnittelijan tai pelaajan yhteistyötä. Esimerkiksi Tanagra-niminen pelikentän suunnittelutyökalu (Smith ym. 2010) luo 2d-pelikenttiä suunnittelijan antamien rajoitteiden mukaan. Työkalu uudistaa pelikenttää samanaikaisesti suunnittelijan päätösten mukaan ja ilmoittaa, mikäli päätöksiä ei voida toteuttaa.

5 Virtuaalisen muistipalatsin rakentaminen

Tässä luvussa käydään läpi, miten kehitysympäristö valittiin ja kerrotaan, kuinka virtuaalisen muistipalatsin rakentaminen toteutettiin. Muistipalatsin luomisessa käytettiin proseduraalista sisällöntuotantoa. Tässä tapauksessa se tarkoittaa muistipalatsin rakentamista algoritmisesti.

Togeliuksen ym. (2011a) ehdottaman proseduraalisen sisällöntuotannon taksonomian mukaan luotu algoritmi on online-tilassa toimiva, tarpeellista sisältöä tuottava, stokastinen sekä generoi ja testaa -sääntöä käyttävä.

Online-algoritmi luo kentän pelin ollessa käynnissä. Tämä asettaa rajoituksia algoritmin nopeuden suhteen, jotta pelaaja ei joudu odottamaan pitkiä aikoja.

Togelius ym. jakoivat algoritmien luodun sisällön kahteen osaan, tarpeellisiin ja vapaaehtoiisiin asioihin. Algoritmi luo huoneet, joiden välillä pelaajan tulee voida kulkea. Täten algoritmin tuottama sisältö on tarpeellista. Algoritmin luoman sisällön tulisi olla virheetöntä, jotta pelaaja voi käyttää kaikkia huoneita. Algoritmia luotaessa tulee miettiä, miten paljon halutaan vaikuttaa lopputulokseen. Tämä vaikuttaminen tapahtuu parametreilla. Mitä tarkemmin lopputulokseen haluaa vaikuttaa, sitä enemmän parametreja tulee käyttää. satunnaislukua käyttäessä, lopputulos on ennalta arvaamaton, täten algoritmi on stokastinen. Algoritmin lopussa se tarkastaa, että huoneet ovat sääntöjen mukaiset ja jokaiseen huoneeseen pääsee kulkemaan. Tätä kutsutaan generoi ja testaa -luokitteluksi.

5.1 Kehitysympäristö

Kehitysympäristön ja pelimoottorin valinnassa painoi eniten sen suosio. Kehitysympäristön suosiota arvioitiin hakutulosten ja tutoriaalien määrällä. Hakutulosten perusteella valittiin Unity3d-kehitysympäristö.

Unityn etuna suosion lisäksi on Asset Store, josta on helppo ostaa tai ladata ilmaiseksi sisältöä pelin kehittämistä varten. Asset Storessa voi myös itse myydä kehittämiään paketteja.

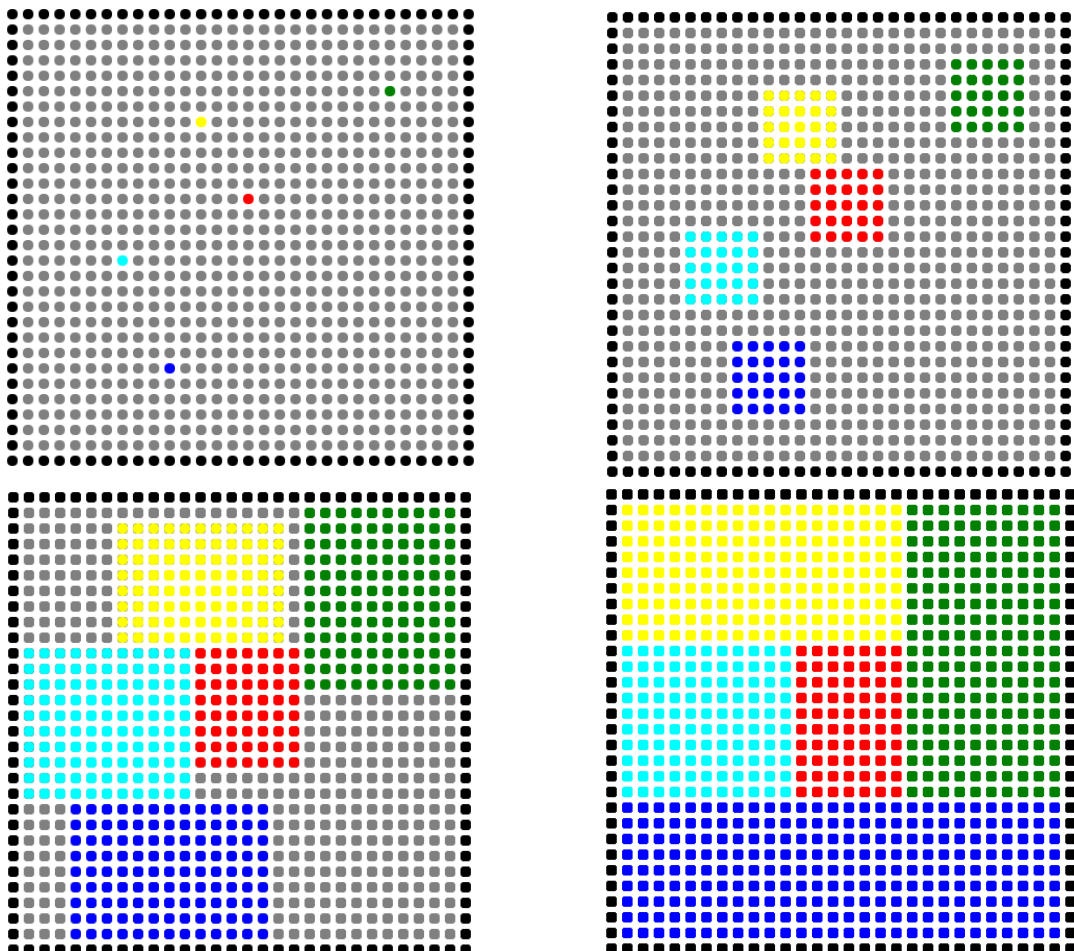
5.2 Algoritmi pohjapiirustuksen luomiseksi

Aluksi pohdittiin omaa toteutusta, jolla ohjelma loisi satunnaisia pohjapiirustuksia. Toteutus ei edistynyt, joten etsittiin verkosta toteutuksia, joista voisi ottaa mallia. Kyseinen ongelma kuuluu proseduraalisen sisällöntuotannon piiriin. Proseduraalinen generointi tarkoittaa, että algoritmi luo erilaisia objekteja. Proseduraalista generointia on käytetty muun muassa pelien vankityrmien, karttojen, aseiden ja kasvillisuuden generointiin (Togelius 2011b).

Pohjapiirustusten luontiin löytyi kaksi algoritmia (Lopes ym. 2010; Merrell ym. 2010). Näistä jälkimmäinen käytti oppimismateriaalina oikeiden asuntojen pohjapiirustuksia, joiden perusteella ohjelma loi bayes-verkon. Jälkimmäinen toteutuksen haastavuuden takia otettiin mallia ensimmäisen artikkelin algoritmista.

Lopes ym. (2010) olivat analysoineet useita algoritmeja pohjapiirustusten tekemiseen ja vertasivat niitä heidän omaan algoritmiin. Heidän oma algoritminsä oli taloudellisempi versio kuin muut läpikäytyt algoritmit.

Heidän algoritminsä alkaa matriisilla, jossa reunat on eristetty arvoilla 0 ja sisemmät pisteet ykkösillä. Tämän jälkeen arvotaan huoneiden lukumäärän verran pisteitä satunnaisesti koordinaatteihin. Nämä pisteet ovat aloituspisteet tuleville huoneille ja käytäville. Seuraavaksi tarkistetaan, etteivät mitkään pisteistä ole vierekkäin. Kun mitkään pisteet eivät ole vierekkäin, ryhdytään laajentamaan pisteitä neliön muodossa. Tätä laajentamista jatketaan, kunnes huoneen seinät törmäävät jonkin toisen huoneen seinään. Tämän jälkeen laajennetaan niitä kylkiä, jotka eivät ole kontaktissa minkään huoneen kanssa. Tätä prosessia jatketaan, kunnes mitään huonetta ei pysty laajentamaan.



Kuva 5. Pohjapiirustus-algoritmin vaiheita

Ratkaisua helpottamaan tehtiin pisteruudukko, mikä helpottaa ongelmatilanteiden ratkaisemista. Pohjapiirustuksen tietorakenteena käytettiin kaksiulotteista taulukkoa. Taulukko sisältää 4-ulotteisia vektoreita. Kolme ensimmäistä vektorin arvoa kuvaavat pisteen paikka avaruudessa ja viimeinen kertoo huoneen numeron. Huoneiden laajentamisen visualisoinnista oli paljon hyötyä. Ongelmatilanteissa visuaalisesta esitystavasta huomasi virheet nopeammin kuin tekstipohjaisesta esitystavasta. Ensimmäinen ongelmista oli huoneen laajentamisen aloittaminen ennen tarkastusta, pystyykö huone laajentumaan joka suuntaan. Tämä loi tilanteita, joissa huone saattoi laajentua jollekin sivulle ainoastaan yhden pisteen verran ja siten luoda huoneisiin erittäin pieniä kulmauksia.

Huoneista muodostuu aina suorakulmioita. Seuraava vaihe olisi muokata algoritmia, jotta huoneet voisivat laajentua toisten huoneiden ympärille. Tämä lisäisi huoneiden erilaisuutta ja mahdollisesti helpottaisi niiden muistamista. Liite 1 sisältää kuvasarjan pohjapiirustuksen muodostumisesta.

Algoritmi valitsee huoneen, jolla on eniten naapurihuoneita. Sen jälkeen etsitään, millä alemman kerroksen huoneella on eniten yhteistä pinta-alaa kyseisen huoneen kanssa. Tästä huoneesta luodaan portaikko yläkerran aulahuoneeseen. Alempana kuva 3 esittää erään algoritmin luoman huoneiston. Yhteisten pinta-alojen laskemista helpottaa huoneiden suorakulmaisuus. Mikäli tulevaisuudessa muuttaa pohjapiirustus-algoritmin luomaan muita kuin suorakulmaisia huoneita, joutuu yhteisen pinta-alan laskemiskoodia muuttamaan.

Tällä hetkellä kerroksesta toiseen pääsee ainoastaan portaita pitkin. Taloihin saisi vaihtelua lisäämällä eri tapoja päästä kerroksesta toiseen. Tällaisia tapoja olisivat esimerkiksi tikapuut, ramppi, hissi tai portaali.

5.3 Huoneiden yhdistäminen

Lopes ym. (2010) kertoo eräästä algoritmista, joka etsii lyhimmat reitit huoneesta toiseen käyttäen A*-algoritmia. Kyseisellä algoritmilla ovien sijaintien löytäminen kestää paljon pidempään kuin satunnaisesti arvotut ovien sijainnit. He eivät kuitenkaan käytä tätä,

vaan he arpoivat oven sijainnin satunnaisesti. Tämä on yksi syistä, miksi heidän algoritminsa on muita nopeampi. Insinööriyönkin algoritmi arpoo oven sijainnin huoneiden välillä. Ohjelmassa käytettiin kaksiulotteista taulukkoa, joka kertoo huoneiden yhteydet toisiinsa. Taulukko on $n * n$ matriisi, jossa n on huoneiden lukumäärä. Sarakkeiden ja rivien indeksit kuvastavat huoneita ja matriisin alkiot ovat näiden huoneiden yhtymäkohtia. Alkion sisältämä koordinaatti kertoo oven sijainnin huoneiden välillä.

Jälkikäteen mietittynä olisi luultavasti ollut helpompaa toteuttaa huoneiden yhteydet suuntaamattomana verkkona. Tällöin olisi ainakin helpompaa tarkastaa, onko huone yhteydessä mihinkään muuhun huoneeseen. Toisaalta verkkoon joutuu erikseen luomaan muuttujan oven sijainnille jokaiselle huoneelle erikseen. Kaksiulotteisessa taulukossa tämän saa suoraan tallennettua soluun. Tämä kaksiulotteinen taulukko on kaikille huoneille yhteinen.

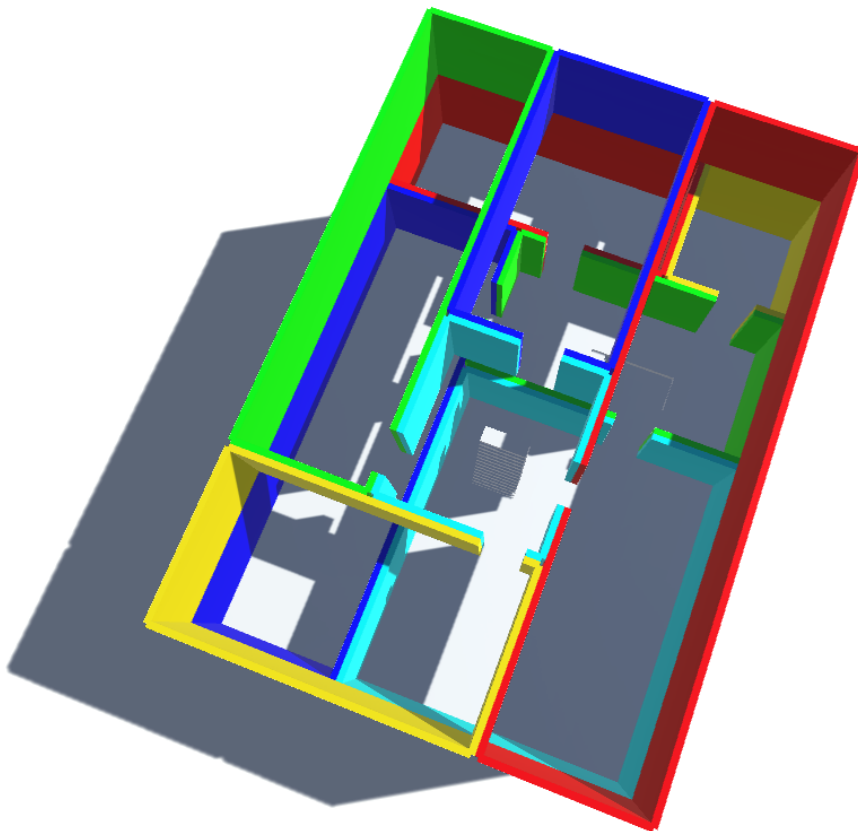
5.4 Algoritmin toteuttaminen

Talon luomiseksi kutsutaan `buildFloor`-metodia kolme kertaa. Metodissa luodaan `Floor`-luokan instanssi ja kutsutaan luokan `createFloor`-metodia. `createFloor`-metodin parametrit ovat `x`-akseli, `z`-akseli, kerroksen syvyys, kerroksen leveys, huoneiden määrä, kerroksen nimi ja kerroksen numero. Kerroksen luonti tapahtuu `do-while`-toistorakenteen sisällä, jotta voidaan tarkastaa, että kerros on luotu annettujen sääntöjen mukaisesti. Tämän perusteella toteutus on etsintään perustuva algoritmi (Togelius ym. 2011a).

Toistorakenteen jälkeen poistetaan turhat objektit. Nämä objektit luodaan turhaan, mutta projektin aikana ei löytynyt koodin osaa tai tilannetta, jossa turhat objektit luodaan. `createFloor`-metodin parametri `houseOffset` arvoa käytetään, kun halutaan luoda useita huoneistoja vierekkäin, parametrin arvoa muutettaessa kerroksen luontia siirretään `x`-akselin suuntaisesti.

```
/**  
private Floor buildFloor()  
{  
    floor = null;  
    do {  
        floor = new Floor();  
        floor.createFloor(22, 0, 14, 20, room,  
level.ToString(), level);  
    } while (!floor.RoomsProperSizes());  
  
    floor.printRoomCoordinates();  
    return floor;  
}  
  
**/
```

Esimerkkikoodi 1. Kerroksen luonti ja tarkastus



Kuva 6. Valmishuoneisto

createFloor-metodissa koordinaatti- ja huoneen koko-parametreja suurennetaan, jotta niitä voidaan paremmin pohjapiirustusten luomisessa. Kun algoritmia käytettiin annetuilla parametreilla, ne olivat niin kaukana toisistaan, että huoneiden seinien välille jäi huomattavat etäisyydet. Parametrien arvoja suurennettaessa pisteiden tiheys kasvoi, ja huoneiden etäisyydet pienenivät. Pisteiden sijainnit tallennettiin kaksiulotteiseen taulukkoon, joka sisälsi vektoreita, jotka sisälsivät tiedon, oliko piste jonkin huoneen sisällä. createFloor-metodin lopussa kutsuttiin RandomizeRoomsUntilApart-metodia.

RandomizeRoomsUntilApart-metodi arpoo jokaiselle huoneelle aloituspisteen. Tämän jälkeen kaikki pisteet käydään vuorotellen läpi ja varmistetaan niiden olevan tarpeeksi kaukana toisistaan. Mikäli tarkistettava aloituspiste on liian lähellä toista pistettä, arvotaan pisteelle uusi sijainti ja tarkistus suoritetaan uudelleen. Pisteiden sijainnin arpominen jatkuu niin pitkään, kunnes piste on sopivalla etäisyydellä tarkistettavasta pisteestä.

Pisteiden arpomisen jälkeen ne sijoitetaan taulukkoon ja ryhdytään laajentamaan aloituspisteitä tyhjiin pisteisiin roomExpansion-metodia kutsumalla. Aluksi metodissa vuorotellaan huoneiden laajentamista niin pitkään, kunnes jokainen huone koskettaa jotakin toista huonetta. Huoneiden kosketukset toisiinsa tallennetaan ja niiden perusteella luodaan arvoalue, josta huoneiden välisten oviaukkojen sijainnit arvotaan.

5.5 Sisustuselementtien luonti

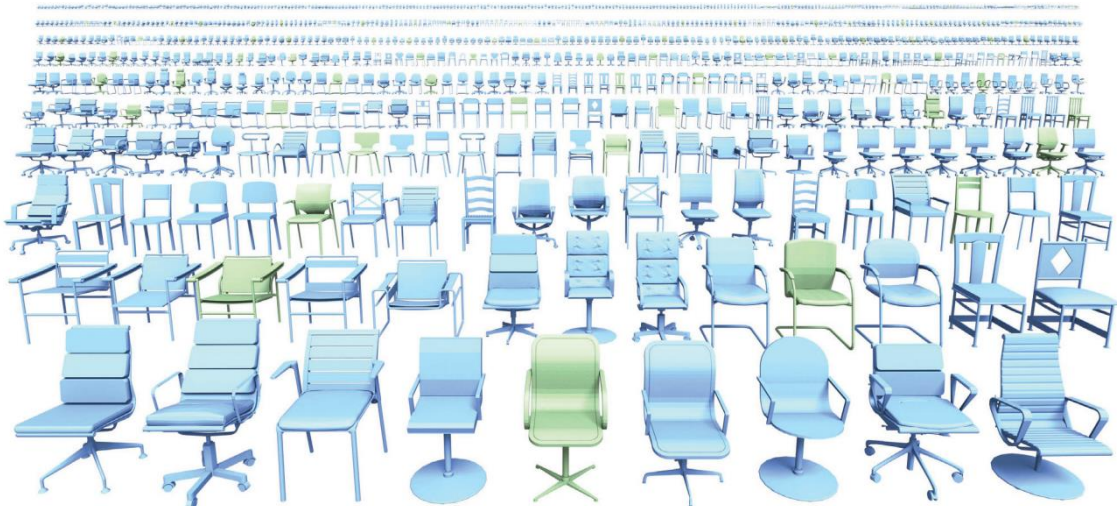
Jotta asunnosta saadaan luontevan näköinen, siellä täytyy olla sisustuselementtejä. Näitä ovat muun muassa kasvit, lamput ja huonekalut. Jotta eri asuntojen sisustukset olisivat erilaisia, täytyy sisustuselementtejä olla paljon. Näiden esineiden luominen yksitellen vaatisi paljon aikaa. Pelialalla pelien esineiden luominen yleisesti on pullonkaulana pelien budjeteissa ja julkaisuaikataulussa (Hendriks ym. 2013). Täten motivaatio kehittää algoritmeja, jotka luovat vähällä työtuntimäärällä laajan kirjon esineitä, on suuri.

Kalogerakis ym. (2012) loivat probabilistisen mallin, joka on suunniteltu kuvaamaan monimutkaisten luokkien esineitä, kuten lentokoneita, huonekaluja ja ajoneuvoja. Prosessissa esimerkiksi mallituolit segmentoidaan. Kuvassa 3 esimerkki segmentoiduista pöydistä. Näiden segmentoitujen tuolien avulla malli oppii, minkälaisia suhteita tuolin eri osien välillä on. Opittujen suhteiden avulla malli kykenee luomaan uudenlaisia ja luonnollisen näköisiä tuoleja.



Kuva 7. Pöytätasohin ja jalkoihin segmentoituja pöytiä

Kalogerakis ym. (2010) esittävät mallilleen kaksi eri sovellutusta, esine-tietokannan suurentamisen ja esineiden suunnittelu rajoitteiden avulla. Tietokannan suurentamisessa



Kuva 8. Mallille annetut tuolit vihreitä, mallin luomat tuolit ovat sinisiä. (Kalogerakis ym. 2010)

malli koulutetaan segmentoiduilla esineillä, jonka jälkeen malli luo kaikki todennäköiset ilmentymät. Kuva 4 on esimerkki tietokannan suurentamisesta. Malli luo vihreistä tuotteista oppimallaan tiedolla sinisen väriset tuolit. Suunnittelu rajoitteiden avulla tarkoittaa interaktiivista suunnittelua. Suunnittelija voi halutessaan valita luokan tietyn komponentin erikseen ja sovellus rakentaa valitun komponentin ympärille vaihtoehtoisia esineitä.

Artikkelissa malli oppi eri tyylisten esineiden, työmaa-ajoneuvojen, olentojen, tuolien, lentokoneiden ja laivojen rakenteita. Eri esineluokkien oppimiseen meni puolesta tunnista 70 tuntiin, joten esineet tulee luoda sovelluksen kehitysvaiheessa.

5.6 Huoneiden kalustaminen

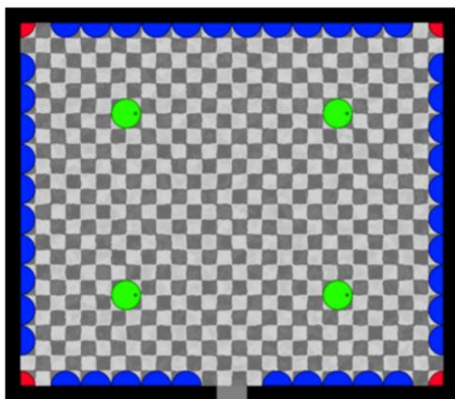
Huoneiden kalustaminen on proseduraalisen sisällöntuoton osa-alue, josta ei vielä löydy paljon kirjallisuutta (Ilcik ym. 2013).

Huoneiden sisustaminen proseduraalisesti vähentää muistajan tekemää työtä. Hänen tarvitsee sijoittaa ainoastaan muistettavat sijainnit eikä tarvitse miettiä merkityksettömien esineiden sijoittamista.

Eräs tekninen raportti käytti petriverkkoja huoneiden sisustamiseen. Petriverkkoja on yleensä käytetty hajautettujen järjestelmien kuvaamiseen, mutta artikkelin kirjoittajat havaitsivat sen soveltuvan myös huoneiden kalustamiseen (Taylor ym. 2010).

Petriverkoissa on kolme osaa, paikat, siirtymät ja merkit. Normaaleissa petriverkoissa kaikki merkit ovat samanlaisia. Artikkelissä käytetään värillisiä petriverkkoja. Värillisissä petriverkoissa merkit pitävät sisällään tietoa. Merkit edustavat ankkureita, joihin sisustuselementit sijoitetaan.

Huoneeseen on sijoitettu koordinaatiston tyyliin tasaisesti ankkuripisteitä. Pisteet voivat olla ominaisuuksiltaan erilaisia riippuen niiden sijainnista. Artikkelissä on kolmen tyyppisiä pisteitä; nurkkapisteitä, seinän vierellä olevia pisteitä ja irrallaan seinistä olevia pisteitä. Seuraavana on visualisointi eri ankkuripisteistä ja niiden sijainneista. Ankkureita voidaan luoda ajonaikana tarvittaessa lisää. Esimerkiksi kun algoritmi sijoittaa pöydän, sen päälle ja ympärille luodaan lisää ankkuripisteitä, joihin voidaan sijoittaa määriteltyjä tavaroita.



Kuva 9. Kuvassa on kolmen tyyppisiä esineiden ankkuripisteitä: kulmapaikkoja, seinän viereisiä paikkoja ja irrallaan seinistä olevia paikkoja.

Harvassa huoneessa tavarat ovat aina täydellisessä järjestyksessä. Siksi artikkelissa on lisätty normaalijakauman mukainen hajonta sisustuselementin suuntaan ja sijaintiin. Täten huone näyttää eletyiltä, eikä ole kuin sisustuslehden mainoskuvasta.

6 Yhteenveto

Sovelluksesta tuli käytettävä prototyyppi. Kaikissa algoritmin tuottamissa huoneistoissa pystyi kulkemaan jokaiseen huoneeseen. Tyypillinen prototyypin käyttötarkoitus olisi esimerkiksi läheisten ja tuttavien puhelinnumeroiden tai syntymäpäivien muistaminen. Numeroiden muistamiseen on matala kynnyks opetusmateriaalin määrän vuoksi. Mikäli haluaisi painaa mieleen jonkin tietotekniikan algoritmin toteutuksen, tarvitsisi ensin muuttaa algoritmissa käytettävät merkit helposti kuviteltaviksi asioiksi.

Huoneiden kalustaminen jäi sovelluksesta toteuttamatta. Seuraavat tehtävät olisivat lähdekoodin refaktorointi ja huoneiden sisustaminen. Refaktoroinnin jälkeen on helpompaa lisätä toiminnallisuuksia.

Huoneiston pohjapiirustusten luominen oli aluksi hankalaa huoneiden koordinaattien muuttuessa joka käynnistyskerralla. Virhetilanteissa huoneiden koordinaatit asetettiin manuaalisesti virheen todentamisen ja korjaamisen helpottamiseksi. Proseduraalisen sisällöntuoton huono puoli on, ettei kaikkia mahdollisia skenaarioita pysty testaamaan. Täten sovelluksessa saattaa ilmetä virheellisiä huoneistoja, vaikka testatessa ei virheitä olisi ilmenekään.

Raportin teon olisi voinut suunnitella tarkemmin. Käytettävä aika olisi tullut jakaa vaiheisiin. Ensin kerätään materiaali ja käydään materiaali läpi, jonka jälkeen kirjoitettaisiin raportti. Nyt tarkemman suunnitelman puuttuessa oli helppo viivästyttää etenemistä siirtymällä edestakaisin vaiheiden välillä.

Lähteet

Bonner, Anthony. 1997. What was Lull up to?. Berliini: Springer Berlin Heidelberg.

Bower, Gordon H.; Clark, Michael C.; Lesgold, Alan M.; Winzenz, David. 1969. Hierarchical Retrieval Schemes in Recall of Categorized Word Lists. *Journal Of Verbal Learning and Verbal Behaviour* 8, 323-343.

Brown, Derren. 2013. Miten mieltä hallitaan. Jyväskylä: Atena Kustannus.

Chua, Elizabeth F.; Schacter Daniel L.; Rand-Giovannetti, Erin; Sperling, Reisa A. 2007. Evidence for a specific role of the anterior hippocampal region in successful associative encoding. *Hippocampus*.

Craik, Fergus I. M.; Tulving, Endel. 1975. Depth of processing and the Retention of Words in Episodic Memory. *Journal of Experimental Psychology: general*, Vol. 104, No. 3, 268-294.

Hawkins, Jeff. Blakeslee, Sandra. 2005. Älykkyys. Helsinki: Edita.

Hendriks, Mark; Meijer, Sebastiaan; VanDer Velden, Joeri; Iosup, Alexandru. 2013. Procedural Content Generation for Games: A Survey. *ACM Transactions on Multimedia Computing, Communications, and Applications*.

Higbee, L. Kenneth. 1979. Recent Research on Visual Mnemonics: Historical Roots and Educational Fruits. *Review of Educational Research*.

Ilcik, M; Wimmer, M. 2013. Challenges and Ideas in Procedural Modeling of Interiors. *Eurographics Workshop on Urban Data Modelling and Visualisation*.

Jenkins, James J.; Russell, Wallace A. 1952. Associative clustering during recall. *The Journal of Abnormal and Social Psychology*, Vol 47(4).

Kalogerakis, Evangelos; Chaudhuri, Siddhartha; Koller, Daphne; Koltun, Vladlen. 2012. A Probabilistic Model for Component-Based Shape Synthesis. *ACM Transactions on Graphics*.

Koenigs, Michael; Barbey, Aron K.; Postle, Bradley R.; Grafman, Jordan. 2009. Superior Parietal Cortex Is Critical for the Manipulation of Information in Working Memory. *The Journal of Neuroscience*.

Kolers, Paul A. 1979. Reading and knowing. *Canadian Journal of Psychology/Revue canadienne de psychologie*, Vol 33(2).

Ledoux, Joseph. 2003. Synaptinen itse. Helsinki: Terra Cognita.

Lopes, Ricardo; Tutenel, Tim; Smelik, Ruben M.; Jan de Kraker, Klaas; Bidarra, Rafael. 2010. A constrained growth method for procedural floor plan generation. Proc. 11th Int. Conf. Intell. Games Simul(Proceedings of the 11th International Conference on Intelligent Games and Simulation).

Maguire, Eleanor A.; Valentine, Elizabeth R.; Wilding, John M.; Kapur Narinder. 2002. Routes to rememberin: the brains behind superior memory. *Nature Neuroscience* 6.

Mattila, Johanna. 2014. <http://opinnot.internetix.fi/fi/muikku2materiaalit/peruskoulu/bi/bi3/06_hermosto/02?C:D=i16E.iSGe&m:selres=i16E.iSGe> <http://creativecommons.org/licenses/by-sa/4.0/>. Luettu 31.10.2016.

McDaniel, Mark A. Pressley, Michael. 1987. *imagery and Related Mnemonic Processes: Theories, Individual Differences and Applications*. New York: Springer-Verlag.

Mintzer, Miriam. Snodgrass, JG. 1999. The picture superiority effect: support for the distinctiveness model. *The American journal of Psychology*.

Merrell, Paul; Schkufza, Eric; Koltun, Vladlen. 2010. Computer-generated residential building layouts. *Proceedings of ACM SIGGRAPH Asia*.

Nadel, Lynn; Hoscheidt, Siobhan; Ryan, Lee R. 2013. Spatial cognition and the hippocampus: the anterior-posterior axis. *Journal of Cognitive Neuroscience*.

Nelson, D. L. 1979. Remembering pictures and words: Appearance, significance, and name. In L. S. Cermak, & F. I. M. Craik (Eds.), *Levels of processing in human memory* (pp. 45-75). Hillsdale, NJ: Erlbaum.

Paivio, Allan. 2006. *Dual Coding Theory and Education. Pathways to Literacy Achievement for High Poverty Children*” Draft chapter.

Rivers, Kimberly A. 2010. *Preaching the memory of virtue and vice*. Turnhout: Brepols.

Rogers, T. B.; Kuiper, N. A.; Kirker, W. S. 1977. Self-reference and the encoding of personal information. *Journal of Personality and Social Psychology*, Vol 35(9).

Slamecka, Norman J.; Graf, Peter. 1978. The generation effect: Delineation of a phenomenon. *Journal of Experimental Psychology: Human Learning and Memory*, Vol 4(6), 592-604.

Smith, Gillian; Whitehead, Jim; Mateas, Michael. 2010. *Tanagra: A Mixed-Initiative Level Design Tool*. *Proceedings of the Fifth International Conference on the Foundations of Digital Games*.

Taylor, Joshua; Parberry, Ian. 2010. *Computerized Clutter: How to make virtual room lived in*. Technical report University of North Texas.

Thomas, Nigel J.T. 2014. Ancient Imagery Mnemonics. Verkkodokumentti <<http://plato.stanford.edu/entries/mental-imagery/ancient-imagery-mnemonics.html>>__Luettu 20.1.2016.

Togelius, Julian; Yannakakis, Georgios N.; Stanley, Kenneth O.; Browne, Cameron. 2011a. Search-Based Procedural Content Generation: A Taxonomy and Survey. IEEE Transactions on Computational Intelligence and AI in Games.

Togelius, Julian; Kastbjerg, Emil; Schedl, David; Yannakakis, Georgios N. 2011b. What is procedural content generation? Mario on the borderline. Proceedings of the FDG Workshop on Procedural Content Generation.

Vann, Seralynne D.; Aggleton, John P.; Maguire, A. Eleanor. 2009. What does the retrosplenial cortex do?. Nature Reviews Neuroscience.

Yates, Frances. 2014. The Art of Memory. London: The Bodley Head.

Pohjapiirustuksen luonti -algoritmin kulku

