

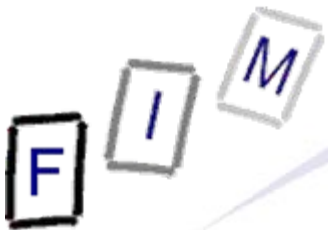
Mag. iur. Dr. techn. Michael Sonntag

Generating cases for E-Learning

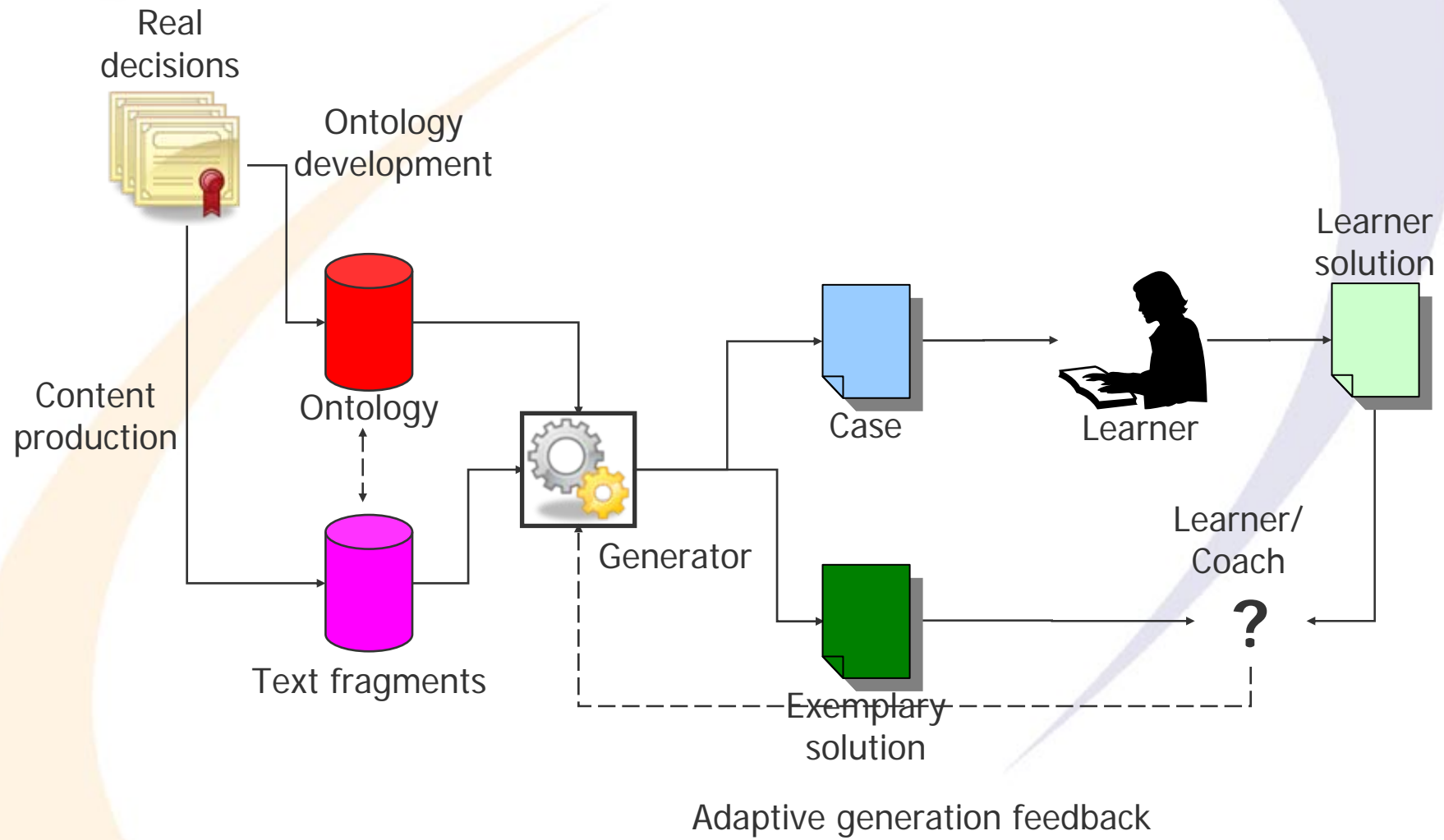
Jurix 2009, Rotterdam, 16.-18.12.2009

Institute for Information Processing and
Microprocessor Technology (FIM)
Johannes Kepler University Linz, Austria

E-Mail: sonntag@fim.uni-linz.ac.at
<http://www.fim.uni-linz.ac.at/staff/sonntag.htm>



- Generating cases for students to solve autonomously
 - Artificial cases, not personalized selection from predefined set
 - E-Learning setting
 - » Self-directed learning by autonomously solving cases
 - » Exemplary solution required for self/peer evaluation
 - Based on ontology describing possible elements and their relations to each other on class level
- Reverse of case modelling for formal analysis
 - Automatic correction would be similar to this!
- Advantages:
 - Generating only such cases, where the result can be determined reliably
 - » But that means also, that not all possible cases are generated...
- Problems:
 - Ontology generation (complex; no procedure for this yet)





Selected domain: The UDRP

- UDRP = Uniform Domain-Name Dispute-Resolution Policy
 - Arbitration procedure for domain names under .com
 - » With modifications also for other top-level domains
 - Practically important: Typically ≥ 2000 cases per year
 - » Reason: Registering a domain = consent to arbitration
 - Requires 4 factors for domain transfer
 - » Domain name is identical/similar to a mark ❶
 - » Bad faith registration ❷ and bad faith use ❸
 - » No legitimate rights or interests of domain name holder ❹
- Restrictions:
 - Implementation: Substantial problems only
 - » Procedural problems not taught in the course it is evaluated in (legal basics for computer scientists)
 - Approach: Only covers factual issues
 - » "Will the described facts be sufficient for subsumption?"
 - » NOT: "Is this rule valid/unconstitutional/higher priority?"



Ontology-based case generation

- Modelling the problem space in a classification hierarchy
 - Class = Problem area
 - Instance = Text fragment describing one possible incarnation of the problem area (positive, neutral, or negative)
- Example:
 - Class "Reimbursement": The domain owner requests some kind of compensation for transferring the domain name
 - Instances
 - » "100 Euro": He requests 100 Euro (→ Legal; keeps domain)
 - » "10.000 Euro": He requests 10.000 Euro (→ Illegal; domain lost)
 - » "Transfer costs": Won't pay for transfer of domain (→ Neutral)
 - » "Link to new site": → Slightly tending towards illegal
- Ontology: Based on problem classification by the WIPO
 - With some modifications
- Text fragments adapted from real decisions



Ontology details

- 4 branches: Domain name, Registration, Use, Legit. rights
 - Each represents one element for the decision of the case
 - Mixing of legal structure and problem space
 - » Possible here because legal structure is trivial!
- Restrictions required: Not every class may appear with each other simultaneously in a single case
 - To ensure factual and legal possibility!
 - » E.g. fragments from "No website under Domain" and "Website similar to one of mark holder" cannot be in the same case
 - Incompatibility on class level
 - » E.g. "selling illegal copies" and "selling genuine products"
 - Incompatibility on instance level
- Multiple inheritance possible:
 - One text fragment contains several problems



Ontology overview

UDRP Protégé 3.3.1 (file:\C:\java\sakai_2-5-2\UDRP\tool\src\java\model\UDRP.pprj, OWL / RDF Files)

File Edit Project OWL Code Tools Window Help

Subclass Explorer CLASS EDITOR

For Project: UDRP For Class: UDRPCase (instance of owl:Class) Inferred View

Property	Value	Lang
rdfs:comment	Ein Fall nach der UDRP.	

Annotations

Asserted Conditions

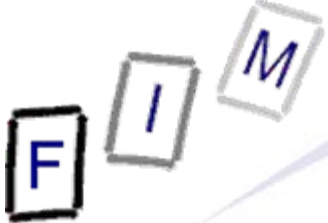
NECESSARY & SUFFICIENT

- owl:Thing
- not (((hasProblem some Commentary) or (hasProblem some Criticism) or (hasProblem some Fan)) and (hasProblem some Parody))
- not (((hasProblem some CommonlyKnown) or (hasProblem some RespondentMark)) and (hasProblem some Derived))
- not (((hasProblem some DisruptCompetitor) or (hasProblem some EvidenceForBadFaithRegistration) or (hasProblem some PriorRegistration)) and (hasProblem some OKRegist))
- hasComplainant max 1
- hasDomainname exactly 1
- hasRegistrationProblem min 1
- hasRespondent max 1
- hasRightsProblem min 1
- not (((hasProblem some Commentary) or (hasProblem some Criticism) or (hasProblem some Parody)) and (hasProblem some Fan))
- not (((hasProblem some Commentary) or (hasProblem some Fan) or (hasProblem some Parody)) and (hasProblem some Criticism))
- not (((hasProblem some CommonlyKnown) or (hasProblem some Derived)) and (hasProblem some RespondentMark))
- not (((hasProblem some Criticism) or (hasProblem some Fan) or (hasProblem some Parody)) and (hasProblem some Commentary))
- not (((hasProblem some Derived) or (hasProblem some RespondentMark)) and (hasProblem some CommonlyKnown))
- not (((hasProblem some EvidenceForBadFaithUsage) and (hasProblem some NoUse))

Disjoints

- Person
- UDRPProblem

Logic View Properties View



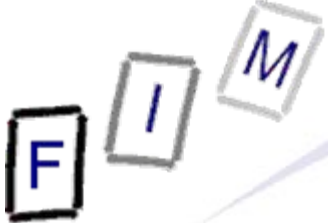
Deriving an exemplary solution

- Each text fragment contains a solution text
 - Explanation for the learner
- Each text fragment has a "fulfilment" value
 - How much it contributes to completing its branch
 - +100 = Alone sufficient; -100 = Alone prevents such finding
 - » Values between: Hints in the appropriate direction
 - Fulfilment values from each branch are summed up
 - Fixed text selected based on ranges of values
 - See example above: Class "Reimbursement"
 - » "100 Euro": Fulfilment = -50
 - Reasonable, but doesn't prohibit a finding of "bad faith use"
 - » "10.000 Euro": Fulfilment = +100
 - Clearly too much, alone sufficient to prove "bad faith use"
 - » "Transfer costs": Fulfilment = 0
 - Doesn't really influence "bad faith use" (happens anyway)
 - » "Link to new site": Fulfilment = +10 (hint towards bad faith)



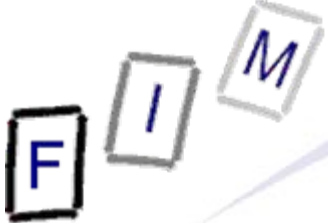
Preconditions for this approach

- All text fragments should be freely combinable
 - Independent from each other on logical and legal level
 - Result: Arbitrary selection possible
 - Restrictions are possible & necessary
 - » This complicates the case generation process and requires a backtracking algorithm for selection
 - » Result: If **specific** classes are required to be present, a solution might be impossible (depends on available instances as well)
- Fulfilment values must be assignable
 - This can be difficult: E.g. -70 or -30 ?
 - » -100 → Be careful! What about other text fragments?
- Fulfilment must be additive:
 - Two text fragments with -50 must be complete proof
 - Two fragments with +10 may not result in a positive finding when occurring together ("synergy")



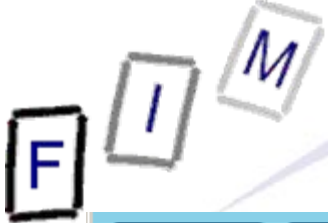
Case completion

- Case is constructed by selecting text fragments randomly
 - Influenced by user & group model and given restrictions
 - » E.g. "I want a case with three 'Use' problems"
 - Created by adding fragments to the case and checking whether the resulting ontology remains valid and consistent
 - » Additional programmatic checks are performed
- Slots are filled from appropriate other fragments or randomly
 - Slots: Domain name, mark, complainant, respondent
 - Some problems (typically from the "Domain name" branch of the ontology) define domain name and mark
 - Complainant and respondent are usually selected randomly, but might be determined by specific text fragments
 - » Example: `wwf.com` → 'WWF' as mark and WWF as complainant
 - Slots are annotated according to grammatical case
 - Alternating person filling: "he", "Mrs. ...", "the complainant" ...



Exemplary implementation

- Java; Integrated into the learning platform Sakai
 - Can also be run separately, but then only the case generation is possible, not the whole learning cycle
- Based on self-assessment
 - For correction the fragments are shown with their solution separately, not as a single text block as before
- Adaptivity features based on individual users and groups
 - Groups: To reduce the cold-start problem
 - Influences problem count and problem selection
 - » According to individual difficulty of classes for person/group
- First feedback from using it in a course with CS students:
 - A bit more variety
 - Self-assessment applet is too complicated/unintuitive



Web UI: Presenting the case

My Workspace Administration Workspace Citations Admin Portfolio Admin **UDRP**

Fallgenerator **UDRP Fallgenerator**

[Entscheidungen](#) [Zurück zur Liste der Fäll](#)

[Stundenplan](#)

[Aktuelles](#)

[Site Stats](#)

[Help](#)

Users present:
Sakai Administrator

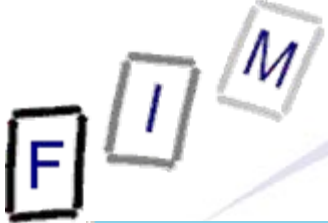
UDRP Fall

Nuka KG ("Mirrorsoft") vs. Acme AG (mirrorsoft.com)

Der Domainname mirrorsoft.com wurde im Jahre 1999 registriert. Im Jahre 2003 kontaktierte ein Rechtsanwalt der Beschwerdeführerin die Beklagte: "Dear Madam/Sir: I am a patent/trademark attorney and have been asked by one of my clients to express to the owner of the "mirrorsoft.com" domain name my client's interest in purchasing the subject URL. I found your name and email address in the domain name records as the administrative contact for the domain. This is intended to be an amicable inquiry. My client has no claim against or conflict with your organization. It would be greatly appreciated if you would reply to this inquiry at your early convenience." Auf diese E-Mail erfolgte keine Antwort. Eine Woche später wurde die Marke zur Eintragung angemeldet. Wieder ein halbes Jahr später wurde unter dem Namen "Mirrorsoft" eine Awareness-Kampagne einer Branchenvereinigung mit einem Budget von 1,2 Millionen US\$ gestartet. Die Domainbesitzerin verwendet für die Registrierung den Anonymisierungsdienst "Moniker". D.h, im Whois-Verzeichnis scheint nicht die Beschwerdegegnerin auf, sondern lediglich die Firma Moniker. Die Acme AG ist der Besitzer des Domainnamens "mirrorsoft.com", wegen dem die Domaininhaberin von der Klägerin verklagt wird.

Ihre Lösung:

Done



Web UI: Self-Assessment

My Workspace Administration Workspace Citations Admin Portfolio Admin **UDRP**

- Fallgenerator
- Entscheidungen
- Stundenplan
- Aktuelles
- Site Stats
- Help
- Users present:
Sakai Administrator

UDRP Fallgenerator

Nuka KG ("Mirrorsoft") vs. Acme AG (mirrorsoft.com)

Auf diese E-Mail erfolgte keine Antwort.
 Eine Woche später wurde die Marke zur Eintragung angemeldet. Wieder ein halbes Jahr später wurde unter dem Namen "Mirrorsoft" eine Awareness-Kampagne einer Branchenvereinigung mit einem Budget von 1,2 Millionen US\$ gestartet.
Die Domainbesitzerin verwendet für die Registrierung den Anonymisierungsdienst "Moniker". D.h, im Whois-Verzeichnis scheint nicht die Beschwerdegegnerin auf, sondern lediglich die Firma Moniker.
 Die Acme AG ist der Besitzer des Domainnamens "mirrorsoft.com", wegen dem die Domaininhaberin von der Klägerin verklagt wird.

IHRE Lösung

Es wurde vor der Markenregistrierung schon der Domainname registriert, daher konnte der Inhaber nichts davon wissen.
 Im Brief wird extra angeführt, dass es **keinen Konflikt gibt, sie also kein Anrecht auf den DN beanspruchen.**
Verdächtig ist die Verwendung eines Anonymisierungsdienstes - deutet auf böswillige Nutzung hin!
 Der Domainname wird übertragen werden!

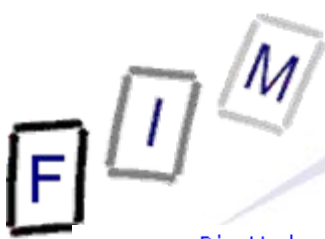
Musterlösung

client has no claim against or conflict with your organization.").
 Die Verwendung eines Anonymisierungsdienstes ist nicht mit der Angabe falscher Daten (Fall Telstra) vergleichbar. Es gibt sehr viele echte und ehrbare Gründe für die Verwendung eines solchen Dienstes, z.B. zur Vermeidung von Spam, Direktmarketing, und Identitätsdiebstahl. Im Falle einer Klage ist dann allerdings der Anonymisierungsdienst

Korrekt
30
Komplett
100

Abbrechen **Lösung speichern**

[Block]



Web UI: Completed case

Die Marke "Pat Benatar" ist zwar nicht durch den Beschwerdegegners eingetragen, aber wurde über lange Zeit von dem Beklagten benutzt und der Beklagte ist auch unter ihr allgemein bekannt.

Prüfen Sie, ob der Domainname auf den Beschwerdeführer übertragen werden wird. Hierzu müssen Sie beurteilen, ob die vier Anforderungen der UDRP (identische/verwechslungsfähig ähnliche Marke, bad-faith use, bad-faith registration sowie kein Recht/berechtigte Interessen) erfüllt werden!

Musterlösung:

In der UDRP sind als berechnigte Interessen auch das Angebot von Dienstleistungen angeführt: Dies ist nicht notwendigerweise kommerziell zu verstehen, sondern auch das unentgeltliche Zurverfügungstellen ist davon umfasst. Das Betreiben eine Fan-Site ist eine solche Dienstleistung für andere Fans und fällt eher unter eine berechnigte Nutzung. Es ist darauf hinzuweisen, dass diese Interpretation unter den Schiedsrichtern umstritten ist: Wird der Markeninhaber dadurch behindert, d.h. kann er selbst nicht unter dem Markennamen auftreten (nicht ähnlich sondern identisch) oder werden über die Fan-Seite auch Einnahmen generiert, so wird größtenteils nicht von einer berechnigten Nutzung ausgegangen. Beides trifft hier jedoch nicht zu.

[Weitere Informationen](#)

Der Domainhaber hat zwar keine registrierte Marke, aber der Markenbesitzer hat den Namen Pat Benatar lange Zeit benutzt und ist darunter bekannt, was ein direkt angeführtes Beispiel für berechnigte Interessen in der UDRP ist. Der bloße Verzicht auf eine Registrierung macht nichts aus, wenn man eine entsprechende Bekanntheit besitzt. Dies ist das Gegenstück zu den "common law" Marken, welche ein Beschwerdeführer besitzen kann, auch ohne dass dafür eine Registrierung besteht.

[Weitere Informationen](#)

Der Domainname besitzt zwar starke Anklänge an die Marke, diese reichen jedoch noch nicht aus. Der Domaininhaber hat ein Recht oder berechnigte Interessen in Bezug auf den Domainnamen. Es gibt einige Anhaltspunkte, die für eine böswillige Registrierung sprechen. Eine böswillige Nutzung des Domainnamens kann nicht ausgeschlossen werden. Da der Domaininhaber ein Recht oder berechnigte Interessen nachweisen konnte, wird der Domainname nicht übertragen.

Ihre Lösung:

Der Domainname ist identisch, da die TLD weggelassen wird. Böswillige Registrierung: Kein Verkaufsangebot, kannte aber offensichtlich den Namen schon! Unklar, ev. ja Böswillige Nutzung: Verbreitet nur positives und Link zur "echten" Homepage. Ist auch tatsächlich ein Fan. Nein Recht oder berechnigtes Interesse: Eher nicht Daher keine Übertragung!



- Research into integrating other dimensions than "problem"
 - Ontology allows multiple classification hierarchies
 - Example: Time (→ More logical ordering of text fragments)
 - Problem: Authoring might become very complex!
- Explicit modelling of case structure
 - Deriving the result not hardcoded but based on a separate model of the logical structure
- More extensive validation
 - Including peer-assessment by other learners
- Integration of more content
 - To improve variety of generated cases
- Improved adaptivity and UI
 - Visualization of user & group models
 - Display of correctness/completeness in completed cases



- Generating cases instead of analyzing them is easier
 - » But not necessarily “easy”!
 - You know the exact structure in advance
 - Only cases possible to solve will be generated
 - » If the model is correct ...
- Working on cases is an important part of legal education
 - Automatic generation allows arbitrary training by learners
- Limitations:
 - At least currently unsuited for complex case structures
 - » Structure might also be generated from "blueprints" and need not be predefined as it is here
 - Not suited to teaching legal argumentation
 - » Only for acquainting learners with interpretation of facts and summarization of the result into an overall result

F I M

Questions?

Thank you for your attention!