

# Standards W3C Voice Browser Activity

Luděk Bártek

Fakulta informatiky  
Masarykova univerzita  
Brno

podzim 2018

# Obsah

- 1 W3C Voice Browser Activity
- 2 VoiceXML

# Úvod

- 1876 - udělen patent na telefon A. G. Bellovi
- WWW
  - 1989 - článek HyperText and CERN (Tim Burns Lee) koloval po CERNU k připomínkám
  - Vánoce 1990 - demonstrován řádkový webový prohlížeč a editor
  - 1991 - všeobecná dostupnost WWW na počítačích v CERNU
  - 1994 - první setkání W3 konsorcia ([www.w3.org](http://www.w3.org))

# Historie

- W3C Voice Browser Working Group
  - založena 1999
  - cíl - návrh standardů umožňujících přístup k WWW pomocí hlasu a telefonu
  - zastoupeny firmy jako:
    - HP
    - Nuance Communications
    - Lucent Technologies
    - Motorola
    - ScanSoft
    - IBM
    - Tellme Networks
    - Vocalocity

# Standardy W3C Voice Browser Activity

- VoiceXML
- Speech Recognition Grammar Specification (SRGS)
- Speech Synthesis Markup Language (SSML)
- Semantic Interpretation for Speech Recognition (SISR)
- Pronunciation Lexicon Specification (PLS)
  - slouží k popisu fonetických informací pro rozpoznávání a syntézu řeči
  - výslovnost zkratk, místních jmen, ...
- Call Control XML (CCXML)
- State Chart XML (SCXML)

## Základní informace

- Jazyk pro popis dialogových rozhraní
- Cíl - přinést výhody webového vývoje a doručování obsahu do interaktivních hlasových aplikací
- vývoj započat 1995 - AT&T Phone Markup Language
- 1998 - konference hostovaná W3C na téma hlasového procházení WWW - předvedeny jazyky PML, VoxML, SpeechML, TalkML, VoiceHTML, ...
- 1999 - založeno VoiceXML Forum - spojení sil při vývoji jazyka pro značkování dialogů
- 2000 - VoiceXML 1.0, krátce na to přijato jako standard W3C
- Aktuální verze:
  - doporučení 2.1 (červen 2007)
  - pracovní návrh 3.0 (prosinec 2010) – zahrnuje podporu pro multimodalitu, čeká se na implementaci.

# Architektura VoiceXML aplikací

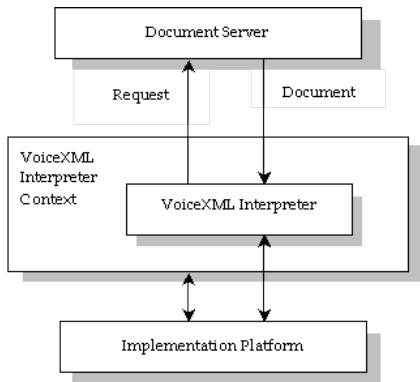


Figure: Převzato ze specifikace VoiceXML 2.0

# Struktura VoiceXML aplikací

- VoiceXML dokument(y)
  - formuláře - konečně stavové automaty.
  - Uživatel se nachází v jednom z konverzačních stavů.
  - Přejechy definovány pomocí URI - odkazují na další krok dialogu.
    - URI - Uniform Resource Identifier
    - jednoznačná identifikace zdroje (souboru, obrázku, ...) na Internetu
    - rozšíření URL (URL je odkaz na soubor, cíl URI nemusí existovat).
  - Dialog končí, pokud tento přechod není definován.
- Dva druhy dialogů:
  - formuláře - definuje proces pro získání hodnot sady položek
  - menu - poskytuje uživateli sadu možností a odkazů na pokračování dialogu



# Struktura VoiceXML aplikací

- Subdialogy
  - obdoba funkcí
  - slouží k opětovnému provádění jisté části dialogu a vrácení získaných hodnot.
- Sezení - začíná v okamžiku, kdy uživatel zahájí interakci s VoiceXML interpretrem a končí, když je komunikace ukončena buď uživatelem, VoiceXML dokumentem nebo kontextem dialogu.
- Aplikace - sada dokumentů, které sdílejí kořenový dokument

# VoiceXML formulář

- Základní komponenta VoiceXML dokumentů.
- Obsahuje:
  - sadu položek
  - deklarace proměnných nepatřících položkám
  - ošetření událostí.
- Základní atribut - id
  - název formuláře
  - lze se pomocí něj na formulář odkazovat
  - musí být unikátní.

# Řízení průběhu dialogů

## Form Interpretation Algorithm - FIA

- Zpracování formuláře - FIA
  - 1 Výběr 1. nezadaného vstupního pole formuláře a jeho zpracování
    - 1 Výběr a přehrání výzev u tohoto vstupního pole (elementy prompt).
    - 2 Získání uživatelských odpovědí, které naplní jednu nebo více položek a nebo vyvolání a zpracování události(žádost o nápovědu, nesrozumitelný vstup, nezadaný vstup, ...).
    - 3 Zpracování sekcí *filled* u všech zadaných položek.
  - 2 Pokud existuje nezadané vstupní pole pokračování krokem 1, jinak zpracování události filled u formuláře.

# Ukázkový VoiceXML formulář

```
<vxml version="2.0" ...>
<form id="hello">
  <block name="hello">
    <prompt>Welcome to the VoiceXML!.</prompt>
  </block>
  <field name="greeting">
    <prompt>Hello.</prompt>
    <grammar root="greeting" src="greeting.grxml"/>
    <noinput>
      <prompt>Tell mi something nice, like hello, hi,
        good day.</prompt>
    </noinput>
  </field>
</form>
```

# Ukázkový formulář

## Pokračování

```
<nomatch>
  <prompt>I didn't understand you, but thanks
  anyway. </prompt>
  <exit/>
</nomatch>
<noinput count="2">
  <prompt>
    You don't want to speak to me. Good bye.
  </prompt>
  <exit/>
</noinput>
</field>
```

# Ukázkový formulář

## Dokončení

```
<filled>  
  <prompt>  
    You said <value expr="greeting"/>  
  </prompt>  
</filled>  
</form>  
</vxml>
```

## Položky formuláře

- Vstupní položky
  - field
  - record
  - transfer
  - object
  - subdialog.
- Vstupním položkám odpovídají proměnné s názvem odpovídajícím hodnotě atributu name příslušné vstupní položky.
- Řídící položky
  - block
  - initial.
- Provádění položky lze omezit pomocí atributu cond.
  - pravdivostní výraz v jazyce ECMAScript s podmínku pro zpracování pole/provádění bloku.

# Element field

- Představuje vstup od uživatele.
- Atributy:
  - name - jméno pole
    - přístup k výsledné hodnotě pomocí stínové proměnné s tímto jménem.
  - expr - případná počáteční hodnota, lze použít výrazy jazyka ECMAScript
  - cond - podmínka nutná pro zpracování vstupu
  - více viz specifikace.



# Element field

## Dokončení

- Obsah:
  - případná výzva s popisem vstupu (element prompt)
  - gramatika - popisuje množinu akceptovatelných vstupů
  - ošetření událostí
    - noinput
    - nomatch
    - filled
    - ...
  - ...

## Ukázka použití elementu field

```
<?xml version="1.0" encoding="UTF-8"?>
<vxml version="2.0" xmlns="http://www.w3.org/2001/v
  <form id="main">
    <field name="name">
      <prompt>Zadejte Vaše jméno</prompt>
      <grammar src="..." type="application/xml+srgs"/>
      <noinput>
        Zadejte prosím Vaše křestní jméno.
      </noinput>
      <nomatch>
        Je mi líto, ale zadané jméno není v kalendáři.
      </nomatch>
    </field>
```

# Ukázka použití elementu field

Dokončení

```
<filled>  
  <submit next="URL aplikace" namelist="name"/>  
</filled>  
</form>  
</vxml>
```

# Element record

- Umožňuje systému nahrát zprávu.
- Lze využít např. pro dialogový záznamník.
- Atributy:
  - name
  - expr
  - cond
  - beep - má-li se před začátkem nahrávání přehrát zvukový signál
  - maxtime - maximální délka nahrávky
  - type - mime-type výsledné nahrávky; musí být podporována VoiceXML platformou
  - ...
- Obsah:
  - případná výzva s popisem vstupu
  - ošetření událostí
    - noinput
    - connection.disconnect.hangup (použití elementu catch).

## Ukázka použití elementu record

```
<?xml version="1.0" encoding="utf-8"?>
<vxml version="2.0" xmlns="...">
  <form id="zaznamnik">
    <record name="zaznam" beep="true" maxtime="30s"
      type="audio/x-wav">
      <prompt> Bohužel zde nikdo není. Po zaznění
        signálu můžete zanechat vzkaz. </prompt>
      <noinput> Bohužel nic neslyším. Zkuste to znovu.
      </noinput>
      <catch event="connection.disconnect.hangup">
        <submit next="http://some.uri.cz/zaznamnik"/>
      </catch>
    </record>
  </form>
</vxml>
```

# Element subdialog

- Slouží k vyvolání dialogu, řešícího dílčí problém.
- Element subdialog.
- Jeden a tentýž subdialog lze volat opakovaně.
- Elementy:
  - subdialog - volání dílčího dialogu
  - param - definice hodnoty parametru
  - filled - kód, který se má provést po návratu z dílčího dialogu.
- Atributy
  - name - jméno volaného dílčího dialogu
  - src - URI dokumentu, který obsahuje kód dialogu.
- Kód subdialogu - formulář, ukončený elementem return.

## Ukázka subdialogu

```
<?xml version="1.0" encoding="utf-8"?>
<vxml version="2.0" xmlns="...">
<form id="demo">
  <subdialog name="greeting" src="#say_hello">
    <param name="param1" expr="'ahoj'"/>
    <filled>
      <prompt> Hodnota subdialogu je
        <value expr="greeting.great"/></prompt>
    </filled>
  </subdialog>
  <filled>
    <prompt>Řekl jste <value expr="greeting.great"/>
    </prompt>
  </filled>
</form>
```

# Ukázka subdialogu

```
<form id="say_hello">
  <var name="param1"/>
  <field name="great">
    <prompt><value expr="param1"/></prompt>
    <grammar root="pozdrav" src="pozdrav.grxml"/>
    <noinput count="2">
      <prompt>
        Na pozdrav jste mi neodpověděl. Nashledanou.
      </prompt>
    </noinput>
  </field>
</form>
```



# Ukázka subdialogu

## Dokončení

```
<nomatch>  
  <prompt>  
    Bohužel jsem Vám nerozuměl, ale stejně dekuji.  
    Nashledanou.  
  </prompt>  
  <return/>  
</nomatch>  
</field>  
<filled>  
  <return namelist="great"/>  
</filled>  
</form>  
</vxml>
```

# Element block

- Obsahuje proveditelný obsah.
- Provádí se pokud:
  - má hodnotu 'undefined' (nebyl dosud navštíven)
  - atribut cond se vyhodnotí jako true.
- Struktura - viz předchozí příklady.
- Atributy:
  - name - jméno bloku
  - expr - iniciální hodnota proměnné formuláře
  - cond - podmínka omezující provádění bloku.

# Element initial

- Typické využití - dialogové strategie se smíšenou iniciativou.
- Umožňuje uživateli zadat více informací naráz.
- Na rozdíl od ostatních vstupních elementů nemůže obsahovat:
  - gramatiku - využívá se gramatika formuláře (viz ukázka na dalším slidu)
    - pokud je uživatelova odpověď gramatikou formuláře rozpoznána, je nutno nastavit hodnotu elementu initial - většinou se používá true
  - filled.
- Dceřiné elementy:
  - výzvy (prompt)
  - ošetření událostí (catch, nomatch, noinput).

## Ukázka použití elementu initial

```
<?xml version="1.0" encoding="utf-8"?>
<vxml version="2.0" xmlns="...">
  <form id="main">
    <grammar src="registrace.grxml"
      type="application/srgs+xml"/>
    <block>
      <prompt>
        Vítejte v registraci předmětů na FI
      </prompt>
    </block>
    <initial name="mixed">
      <prompt>
        Zde můžete zadat, jaké předměty s jakým
        ukončením si chcete zaregistrovat
      </prompt>
```

# Ukázka použití elementu initial

## Pokračování

```
<noinput>  
  Řekněte něco jako Předmět PB095 na zkoušku  
</noinput>  
<noinput count="2">  
  Dobře zkusíme to postupně.  
  <assign name="mixed" expr="true"/>  
  <reprompt/>  
</noinput>  
<nomatch>  
  Můžete si zaregistrovat předměty PB095, PB125,  
  PB162 s hodnocením zkouškou nebo zápočtem.  
</nomatch>
```

# Ukázka použití elementu initial

## Pokračování

```
<nomatch count="2">  
  Dobře zkusíme to postupně.  
  <assign name="mixed" expr="true"/>  
  <reprompt/>  
</nomatch>  
</initial>  
<field name="predmet">  
  <grammar src="registrace.grxml\#predmet"/>  
  <prompt>Zadejte kód předmětu</prompt>  
  <nomatch>  
    Zatím si lze zaregistrovat předměty PB162,  
    PB095, PB125.  
  </nomatch>
```

# Ukázka použití elementu initial

## Pokračování

```
<nomatch count="3">  
  Bohužel se nám zadávání nedaří. Nashledanou.  
  <exit />  
</nomatch>  
<noinput count="3">  
  Jelikož jste nic nezadal, tak se s Vámi loučím.  
  <exit />  
</noinput>  
</field>  
<field name="ukonceni">  
  <grammar src="registrace.grxml#ukonceni"/>  
  <prompt>Zadejte požadované ukončení.</prompt>
```

# Ukázka použití elementu initial

## Pokračování

```
<nomatch>Předměty lze ukončit zkouškou  
nebo zápočtem.  
</nomatch>  
<nomatch count="3">  
  Bohužel se Vám zadávání nedaří, zkuste to  
  klasicky na adrese is.muni.cz. Nashledanou.  
</nomatch>  
<noinput>Zadejte, zda chcete předmět ukončit  
zkouškou nebo zápočtem.  
</noinput>  
<noinput count="3"> ... </noinput>  
</field>
```



# Ukázka použití elementu initial

## Dokončení

```
<filled>
  <prompt>
    Provádím registraci předmětu s kódem
    <value expr="predmet"/>
    a ukončením <value expr="ukonceni"/>.
  </prompt>
</filled>
</form>
</vxml>
```

## Další informace o VoiceXML

- Popis na W3C Voice Browser Activity  
(<http://www.w3.org/TR/voicexml20/> doplnění  
změn na <http://www.w3.org/TR/voicexml21/>)
- [www.voicexml.org](http://www.voicexml.org) (<http://www.voicexml.org>)
- Šimek, Richard - Tutoriál jazyka VoiceXML (bakalářská  
práce FI), 2005
- ...