Ontology for Transcription of ATC Speech Commands of SESAR 2020 Solution PJ.16-04

Hartmut Helmke[1], Michael Slotty[2], Michael Poiger[3], Damián Ferrer Herrer[4], Oliver Ohneiser[5], Nathan Vink[6], Aneta Cerna[7], Petri Hartikainen[8], Billy Josefsson[9], David Langr[10], Raquel García Lasheras[11], Gabriela Marin[12], Odd Georg Mevatne[13], Sylvain Moos[14], Mats N. Nilsson[15], Mario Boyero Pérez[16]

[1,5] Institute of Flight Guidance, German Aerospace Center (DLR), Braunschweig, Germany,
[2] ATM Simulator Centre, DFS, Langen, Germany,
[3] Frequentis, Vienna, Austria,
[4,11] CRIDA, Madrid, Spain,
[6] NATS, Whiteley, Fareham, United Kingdom,
[7,10] Air Navigation Services of the Czech Republic, Jeneč,
[8] Integra, Copenhagen, Denmark,
[9,15] LFV, Stockholm, Sweden,
[12] Romatsa, Bucharest, Romania,
[13] Avinor, Oslo, Norway,
[14] ATM Innovation Lab, Thales Air Systems, Rungis, France,
[16] Indra, Madrid, Spain, no with Eurocontrol

firstname.lastname@{[1,5]dlr.de, [2]dfs.de, [3]frequentis.com, [6]nats.co.uk, [7],ans.cz, [9]lfv.se, [12]romatsa.ro, [14]thalesgroup.com}, [4]dfherrer@e-crida.enaire.es, [8]pha@integra.dk, [10]davidlangr@europe.com, [11]rglasheras@e-crida.enaire.es, [13]odd.georg.mevatne@avinor.no, [15]mats.n.nilsson@lfv.nuac.eu, [16]mario.boyero-perez@eurocontrol.int

***Abstract*—Nowadays Automatic Speech Recognition (ASR) applications are increasingly successful in the air traffic (ATC) domain. Paramount to achieving this is collecting enough data for speech recognition model training. Thousands of hours of ATC communication are recorded every day. However, the transcription of these data sets is resource intense, i.e. writing down the sequence of spoken words, and more importantly, interpreting the relevant semantics. Many different approaches including CPDLC (Controller Pilot Data Link Communications) currently exist in the ATC community for command transcription, a fact that e.g. complicates exchange of transcriptions. The partners of the SESAR funded solution PJ.16-04 are currently developing on a common ontology for transcription of controller-pilot communications, which will harmonize integration of ASR into controller working positions. The resulting ontology is presented in this paper.**

Keywords—Automatic Speech Recognition (ASR); CWP HMI; Transcription; Controller Command; Ontology; SESAR; PJ.16-04

I. Introduction

A. Problem

Speech Recognition applications have dramatically improved over the last decade. Familiar examples include; Apple Siri®, Amazon Alexa, Google Assistant and Microsoft Cortana. The integration of Automatic Speech Recognition (ASR) in ATC training started in the late 80s [1]. Nowadays enhanced ASR systems are used in ATC training simulators to replace pseudo pilots and to improve the utilized capacity of the simulator infrastructure (e.g., FAA [2], DLR [3], MITRE [4] and DFS [5]). Today ASR applications go beyond simulation and training. ATC events can automatically be detected in order to assess controller workload, i.e. ASR is used to get more objective feedback of controllers' workload [6], [7]. Chen and Kopald used speech recognition to build a safety net for airport surface traffic to avoid aircraft using a closed runway [8]. Most recently, they presented an approach to detect pilot read back errors [9].

Although the vocabulary in controller-pilot communication is quite limited and phraseology is restricted, recognition rates are still far from being perfect. One reason is the limited amount of available system training data: The MALORCA project [10] is based on only 50 hours of data. Chen and Kopald [9] reported a learning data set size of 130 hours. Google on the other hand has used more than 200,000 hours of training data to improve the speech recognition engine. Tüske et al. train a Neural Network based language model with **3.1 billion** (!) words [12] for understanding English broadcast news. Xiong et al. use 2,000 hours to compare human transcription performance with ASR performance [13]. Chen et al. use 2,100 hours of training for English conversational telephony understanding [14]. Thousands of hours of ATC communication are recorded every day. However, the transcription of these data sets is the expensive part. Although MALORCA tries to automate this process, a basic set of manual transcriptions is always needed. Exchange of previously transcribed data between different projects is challenging. The sequence of spoken words is the easier part and, therefore, cheaper part. MALORCA project has shown that transcription of the semantics (i.e. the ATC relevant concepts) is the challenge [22]. What are the relevant concepts of "lufthansa two alpha good morning altitude four thousand feet on qnh triple nine"? Are these DLH, 2A, greeting, 4000 feet, QNH, 999? Or are greeting and feet irrelevant? Or is DLH2A the concept? Or is DLH132A the correct callsign, because DLH2A is just an abbreviation? As we show in section II, many different approaches currently exist.

B. Solution for an agreed command transcription ontology

The SESAR funded solution PJ.16-04 of the project Controller Working Position Human Machine Interface (CWP

© 2018 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works

The work is partly funded by SESAR Joint Undertaking (Grant Number 734141).

HMI) is developing a so-called Ontology for Command Transcription. Ontologies define abstract concepts and the relations between them. They are mostly used to exchange knowledge in digital form between computer programs. The knowledge can comprise both general and expert knowledge. The term ontology establishes itself in computer science (especially in the sectors of semantic web and Natural Language Understanding).

C. Other Benefitting Applications

A developed ontology not only supports data exchange between different stakeholders, i.e. does not only increase the size of training data, but is also needed in applications with automatic pseudo pilots. The relevant concepts need to be transferred into commands for the simulator. The “good morning” in the introductory example is irrelevant, DLH2A and the altitude value are important. The QNH value is normally ignored. In an automatic simulation environment or in a training simulator for controllers the read back of the pilots is very important..

If we are able to extract the semantics not just from the controller, but also from the pilot read back, then we can also detect read back and hear read back errors. It is of course not sufficient only to check if the pilot repeats exactly the controller’s word sequences as this *never* happens. A read back for the above DLH2A example could be “four thousand zero nine nine nine lufthansa one three two alpha” or “triple nine for four thousand feet two alpha”. Chen and Kopald also showed that some commands are implicitly read back, especially during taxi clearances [9]. This indicates that concept extraction (command transcription) is not sufficient for read back cross checking, but it is an indispensable precondition.

ASR will play a decisive role in PJ 16-04. Austro Control had been using VAS, in the Vienna ATM System, but in 2007, it was decided VAS would be replaced by a completely new Air Traffic Management (ATM) system. Ireland, Sweden, Denmark, Croatia, and Austria mutually agreed on the common Thales TopSky ATC system [11]. While VAS used paper strips to provide the air traffic controllers (ATCos) with all required flight data, TopSky ATC already operates a paperless system and thus important information is now presented on the radar screen (see Fig. 1 and 2 taken from Thales Shape HMI prototyping system).

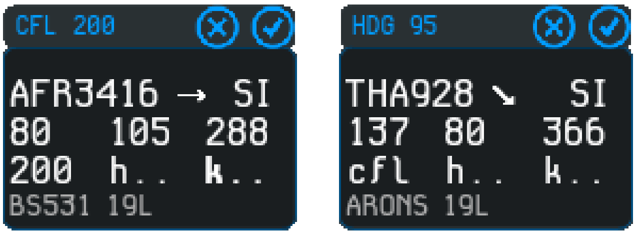


Figure 1. Recognized Commands integrated (as a shortcut) in Radar Label.

This is mostly challenging for the approach control units, which need to input the highest number of instructions in narrow timeframes. Input is predominantly provided manually by mouse, as well as with keyboard inputs. As shown in [30] ASR could be a solution to reduce controllers’ workload significantly. The input needed for the radar labels could be directly extracted from the controller-pilot communication. Again, we need to define the relevant concepts of the utterances.

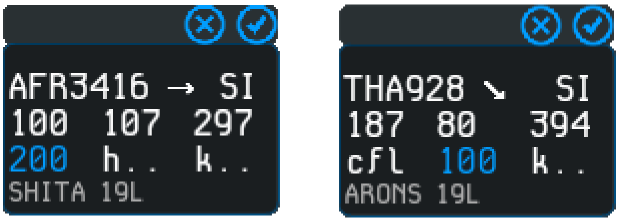
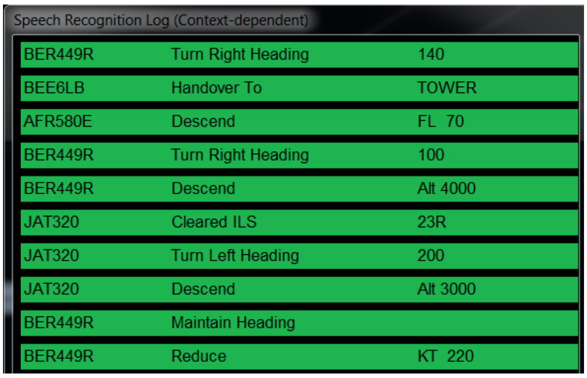


Figure 2. Recognized Commands integrated inside Radar Label.

Another application of ASR in ATC may be to visualize the sequence of all given commands of the last five minutes (Fig. 3) or, of all commands given to one specific aircraft.



Speech Recognition Log (Context-dependent)

BER449R	Turn Right Heading	140
BEE6LB	Handover To	TOWER
AFR580E	Descend	FL 70
BER449R	Turn Right Heading	100
BER449R	Descend	Alt 4000
JAT320	Cleared ILS	23R
JAT320	Turn Left Heading	200
JAT320	Descend	Alt 3000
BER449R	Maintain Heading	
BER449R	Reduce	KT 220

Figure 3. Command History showing Callsign, Type, and Value.

Furthermore, a beneficial application is to characterize sectors and detect workload [7] based on the commands given and how often they are used. Table 1 shows the share of altitude commands for eight Spanish sectors from Madrid and Barcelona. Upper sectors (last letter a “U”) present a lower percentage of altitude change commands than lower sectors (L as last letter of name) (5-15% vs 17-24%).

TABLE 1: ALTITUDE COMMAND DETECTION CHANGE

Sector Name	*Hours analyzed*	*Altitude commands per hour*	*Total commands*	*Altitude commands*
LECMPAL	50	12.95	647	16.6%
LECMPAU	50	4.30	216	4.8%
LECMASL	50	8.40	419	21.2%
LECMASU	50	4.40	220	7.0%
LECMZML	50	18.55	927	19.0%
LECMZMU	50	7.25	363	9.1%
LECBP1L	50	27.90	1395	24.0%
LECBP1U	50	12.10	604	14.8%

Data has been derived from CRIDA ASR data base consisting of 590 hours of transcribed data.

Even if ASR is not used to replace mouse and keyboard input, ASR could cross check extracted concepts against the controllers' input into an electronic flight strip system or label. ASR output can also be used for cross checking against Mode S output. In all described application cases of this section a command transcription, i.e. understanding the sequence of words of controller pilot communication, is needed as precondition.

Related work is presented in the following chapter; the developed ontology follows in chapter III; chapter IV compares ASR and our ontology to CPDLC. First applications of the ontology within SESAR exercise 16-04 follow in chapter V before we present our conclusions.

II. Related Work

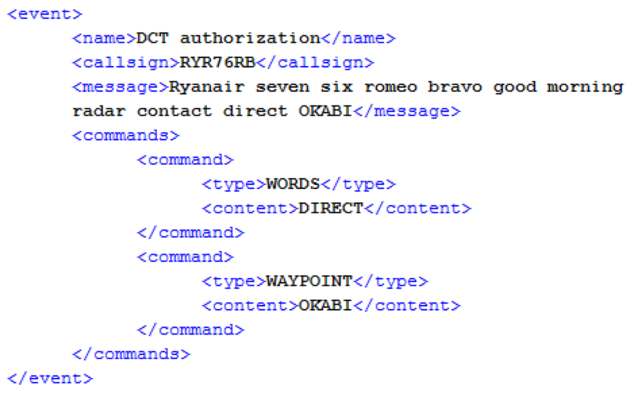
Nguyen and Holone [15], [16] propose 10 classes to replace word sequences with their corresponding class labels: 1) callsign, 2) unit-name, 3) fix, 4) number, 5) letter, 6) greeting, 7) non-verbal articulations (ah, yeah, aha, etc.) and the three minor classes 8) directions (left, right, etc.), 9) position (above, below …), 10) unit (feet etc.). Johnson et al. [17] propose a keyword and value representation in JSON format [18]. Keywords are e.g. Callsign, ToFix, FlightLevel, and Altimeter.

Saarland University and DLR created in the AcListant® project [19] ontology, which only consists of four elements: 1) callsign, 2) command type, 3) commanded value and 4) unit [20], [21]. Callsign and command type are mandatory. Unit is only used for altitude commands, i.e. only flight level and feet are used. The utterance "lufthansa two alpha altitude four thousand feet on qnh one zero one four reduce one eight zero knots or less turn left heading two six zero" is transcribed into "DLH2A ALTITUDE 4000 ALT DLH2A REDUCE_OR_BELOW 180 DLH2A TURN_LEFT_HEADING 260". The callsign is repeated for each command in an utterance consisting of multiple commands. "Reduce or less" e.g. is transformed into a new command type REDUCE_OR_BELOW. More than 30 command types were supported. The approach reaches its limits in the MALORCA project [22] when it was extended for command transcription for live traffic in Vienna and Prague approach. Departure and overflight traffic had to be included there. More and more command types were needed (e.g. QNH, INFORMATION, REPORT_SPEED, EXPECT_RUNWAY) and the necessity to transcribe conditional clearances occurred [23].

Starting in 2002, NATS conducted a four-year project to look at the possible applications of ASR within the London Terminal environment [24]. Several ontologies were proposed, initially based on a Statistical Language Model. This was to overcome the expanded phraseology demanded by NATS procedures. Latterly, refinements were made utilising a grammar model, which showed remarkably more promise but limitations of the technology at the time restricted the amount of phraseology context data which could be used to support this. At project closure, the ontology encompassed five elements consisting of: 1) callsign, 2) standard (International phraseology) type, 3) non-standard (i.e. NATS based procedures) type, 4) type value and, 5) type currency/unit (e.g. feet, degrees). As part of NATS research, a Frequentis test system and several commercial off-the-shelf products, including Nuance 8.5 and Telisma, have been tested.. With an average of over 600 utterances per controller per hour at Heathrow approach, a recognition rate above 80% was needed to warrant operational use, however, none of the evaluated speech recognition engines could meet this. NATS discovered that the lack of sufficient context data is further exacerbated by native English speakers lapsing into conversational terms outside of standard ATM phraseology. It was also noted that there were a lack of universal standards for quick and economical application development.

In 2011, DFS started the introduction of Voice Recognition and Response for the DFS controller training to replace simulation pilots. The simulator integrates the BBN HARK AVOKE finite state speech recognition system. It was started to implement successively the DFS standard phraseology. After more than six years of using speech recognition in Pre-On-The-Job training and different types of proficiency training like fallback system refresher training, cross training, etc., DFS is continuing their effort to use speech recognition in more areas of training across all field units and the DFS Academy.

Since 2008, ENAIRE, EML, and CRIDA have been working together on an ASR prototype called VOICE. The speech recognition module of the prototype produces a transcription of the audio segment, either in English or in Spanish. From the transcription, callsigns and events are identified, according to defined grammar rules, which are used to create the final XML output file. An example of the contents of the output XML file is shown in Fig. 4.



```
<event>
      <name>DCT authorization</name>
      <callsign>RYR76RB</callsign>
      <message>Ryanair seven six romeo bravo good morning
      radar contact direct OKABI</message>
      <commands>
            <command>
                  <type>WORDS</type>
                  <content>DIRECT</content>
            </command>
            <command>
                  <type>WAYPOINT</type>
                  <content>OKABI</content>
            </command>
      </commands>
</event>
```

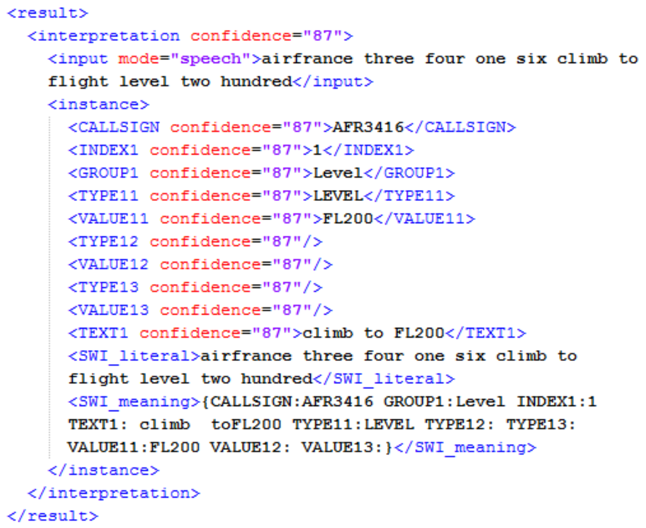
Figure 4. Example of controller events in XML output file.

Frequentis have been investigating applications of ASR in ATC for more than 10 years, beginning with the user-friendly integration of ASR into the electronic strip system. Based on the experience gained during these years, Frequentis together with DFS are now tackling the next challenge; to investigate the use of ASR for multi remote tower operations to improve situational awareness, taking the communication of the controller and the pilot into account. There is also a special focus on a seamless integration of this service into the Human Machine Interface. To tackle this challenge it is very important to have a clear understanding of the voice information transmitted and the amount of possibilities for specific situations occurring to train the ASR service in this regards.

One clear example of this challenge is the large volume of surplus language used that is not standard phraseology or required for flight information. One NATS terminal approach sector found that more than 20% of utterances were passed for information or for courtesy. The transmission, “speedbird two one maintain speed until advised any delay will be less than ten” includes information that is outside standard phraseology and challenging for an ASR service.

INDRA is continuously enhancing an autonomous speech recognizer. The main objective of the ASR is for training of ATCos by simulating a real scene through different instructions that are preconfigured. The CWP, with integrated ASR, supports the introduction of commands using air ground data link tools. Therefore, without the need to establish voice communications with the aircraft, helping the controller to reduce workload and improve efficiency. Currently, the recognized phraseology includes more than 50 words based on ICAO’s standard. Hours of speech have been recorded and analysed to improve the accuracy of the recognizer taking also into account the different controllers’ accents.

Since 2013, Thales has been working on a platform called ‘Shape,’ integrating various modalities including an ASR module based on the Recognizer 10 engine from Nuance Communications. The voice recognition engine is assisted by a hypotheses engine developed by Thales. Given the audio segment and a grammar file generated by the hypotheses engine, the voice recognition engine produces a transcription of the audio segment. The Air Situation Display (ASD) uses the transcription as an input for clearances. An example of the contents of the output XML file is shown in Fig. 5.



```
<result>
  <interpretation confidence="87">
    <input mode="speech">airfrance three four one six climb to
    flight level two hundred</input>
    <instance>
      <CALLSIGN confidence="87">AFR3416</CALLSIGN>
      <INDEX1 confidence="87">1</INDEX1>
      <GROUP1 confidence="87">Level</GROUP1>
      <TYPE11 confidence="87">LEVEL</TYPE11>
      <VALUE11 confidence="87">FL200</VALUE11>
      <TYPE12 confidence="87"/>
      <VALUE12 confidence="87"/>
      <TYPE13 confidence="87"/>
      <VALUE13 confidence="87"/>
      <TEXT1 confidence="87">climb to FL200</TEXT1>
      <SWI_literal>airfrance three four one six climb to
      flight level two hundred</SWI_literal>
      <SWI_meaning>{CALLSIGN:AFR3416 GROUP1:Level INDEX1:1
      TEXT1: climb  toFL200 TYPE11:LEVEL TYPE12: TYPE13:
      VALUE11:FL200 VALUE12: VALUE13:}</SWI_meaning>
    </instance>
  </interpretation>
</result>
```

Figure 5. Example of recognition result in XML output file.

We have seen that different European stakeholders have common objectives when integrating ASR into ATM applications, but their output formats are still very different.

III. Developed Ontology

After having described the need for a common understanding of command transcription and related work, we are now describing the ontology developed within SESAR PJ.16-04 solution. The main elements of the ontology are: callsign and instruction. Fig. 6 shows that an instruction always consists of a command (darker green part is mandatory) and one or more optional (orange) conditions described in detail later. A command is composed of a type, one or more values and a unit (e.g. FL or ft or none). Then an optional qualifier follows (e.g. LEFT, RIGHT, OR_LESS, BELOW).

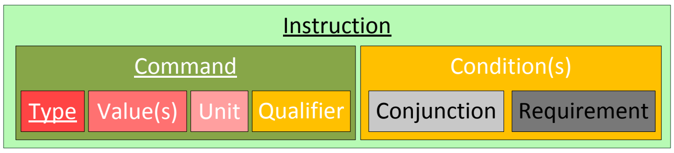


Figure 6. Elements of an instruction of a clearance.

An utterance may consist of multiple instructions for the same callsign or even for different callsigns (separated by “break, break”). We always add the callsign to the instruction (or NO_CALLSIGN) independent of being repeated by the controller. Not all command types require a value or a unit, but all of them need a type. The type consists of one or two words: “CLEARED TOUCH_GO” is a two word example. “STRAIGHT_IN_TURN” or “DESCEND” are one word examples.

A. *Commands for Vertical Guidance*

A vertical command may be a climb or a descend command (see Fig. 7) or just the altitude or flight level value that is said, e.g. “speedbird alpha bravo flight level eight zero”. In that example we can derive in most cases whether it is a climb or descent from the context (inbound/outbound or current altitude), but our approach is to transcribe just what we can derive from the current utterance alone. This is different to CPDLC [25] or ontologies used to guide ATC simulators: the ALTITUDE command type e.g. is lost, because it is transformed to CLIMB or DESCEND. An OR_BELOW qualifier is also not useful for a simulator.

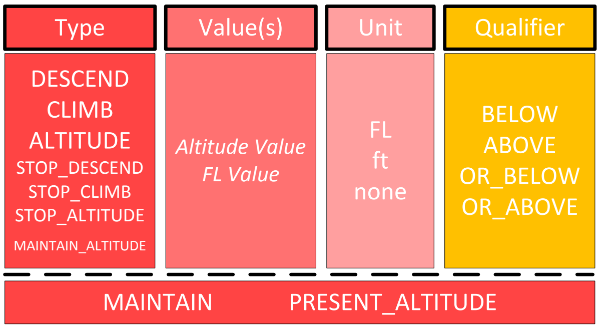


Figure 7. Elements of a Vertical Instruction (with optional orange Qualifier).

Fig. 7 shows that we transcribed stop commands and maintain commands as separate commands. We also use MAINTAIN_ALTITUDE for “lufthansa two alpha maintain flight level one one zero”. We get “DLH2A MAINTAIN_ALTITUDE 110 FL”, because we want to avoid the need for a separate command type for flight levels and for altitudes. “two alpha maintain flight level,” however, is transcribed with

"DLH2A CONTINUE_PRESENT_ALTITUDE" which will ease implementation of transcription passing. We can also model whether the unit (FL resp. ft) was said / recognized or no unit was said / recognized (none).

B. *Commands for Horizontal Guidance*

Fig. 8 and Fig. 9 detail the horizontal commands. Wet give two examples: "csa one yankee golf left turn zero four zero" results in "CSA1YG HEADING 040 LEFT". "austrian golf golf one proceed delta lima four five five and adama" results in "AUAGG1 DIRECT_TO DL455 ADAMA", i.e. DIRECT_TO is followed by one or more waypoints. It is not split into the two separated instructions "AUAGG1 DIRECT_TO DL455" and "AUAGG1 DIRECT_TO ADAMA".

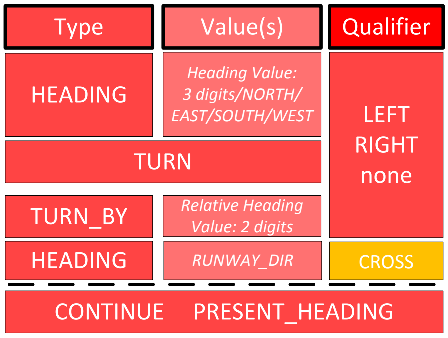


Figure 8. Elements of a horizontal command (part 1).

We see that (1) no units are provided for directions commands and (2) the qualifier is mostly mandatory, i.e. "heading zero two zero" is transcribed as "HEADING 020 none." "Turn left by twenty degrees" results in "TURN_BY 20 LEFT". "MAINTAIN_HEADING" is always without qualifier.

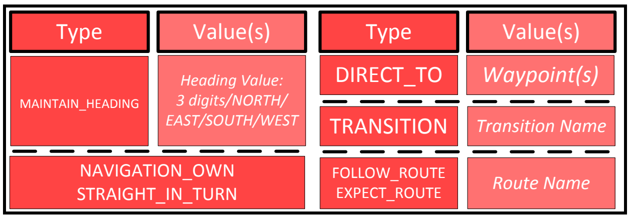


Figure 9. Elements of a horizontal command (part 2).

C. *Commands for Taxi Guidance*

Fig. 10 details the taxi command types of a tower controller. We provide some examples in Table 2 to Table 5. We start with the (transcribed) controller utterance followed by the resulting command transcription and include additional text explaining our decisions if relevant.

TABLE 2: LINEUP-BEHING COMMAND EXAMPLE

Utterance	portugal one lima whiskey behind the landing airbus three twenty over the threshold lineup runway one three right
Transcription	TAP1LW LINEUP BEHIND 13R AIRBUS320
Explanation	"over the threshold" is not transcribed in our ontology.

TABLE 3: LINEUP COMMAND EXAMPLE

Utterance	portugal one lima whiskey debrecven tower line up runway zero five right and wait
Transcription	TAP1LW LINEUP 05R
Explanation	"wait" is not transcribed to "HOLD" command. The stopping after lining up is implicit.

TABLE 4: TAXI TO COMMAND EXAMPLE

Utterance	swiss three five four thank you continue taxi to your parking
Transcription	SWR354 TAXI TO STAND
Explanation	Although the controller says "continue taxi" we transcribe this as "TAXI TO". We use "CONTINUE TAXI" if said without destination. STAND is a TX-Point name, which needs to be defined airport specifically in a data base.

TABLE 5: VACATE TO COMMAND EXAMPLE

Utterance	swiss three five four vacate runway via taxiway bravo and golf
Transcription	SWR354 VACATE VIA B G
Explanation	"please vacate via taxiway bravo expedite" would results in "VACATE VIA B EXPEDITE".

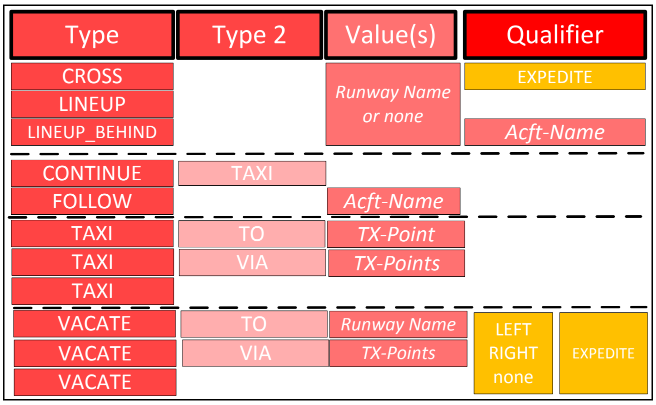


Figure 10. Elements of a taxi command.

D. *Additional Command Type Categories*

Besides vertical, horizontal and taxi command type categories, we distinguish between

- speed commands (e.g. HIGH_SPEED_APPROVED, REDUCE),
- air holding (e.g. HOLDING, ORBIT) and ground hold commands (e.g. HOLD_AT, HOLD_SHORT),
- vertical speed commands (e.g. RATE_OF_DESCENT),
- changing frequency commands (e.g. CONTACT),
- report and request commands,
- landing clearances (e.g. CLEARED ILS RW26, CLEARED NDB, INTERCEPT_LOCALIZER) and their cancelation (e.g. CANCEL TOUCH_GO),
- taxi commands (e.g. LINEUP, VACATE),

- clearance delivery commands (e.g. PUSHBACK, STARTUP),
- information commands (e.g. QNH, WINDSPEED),
- VFR clearances (e.g. LEAVE_CTR, JOIN_TRAFFIC_CIRCUIT), and
- miscellaneous commands (e.g., CALL_YOU_BACK, CORRECTION).

We provide examples for command transcription of the additional command type categories to explain our ontology in Table 6 to Table 9. We start with the (transcribed) controller utterance followed by the resulting command transcription and include additional text explaining our decisions where relevant.

TABLE 6: SPEED COMMAND EXAMPLE

Utterance	monarch two nine zero eight fly speed three five zero knots
Transcription	MON2908 SPEED 350 kt
Explanation	The controller says “niner” and “tree”, but in word transcription we use “nine” and “three” and completely small letters. kt resp. MA for (mach numbers) is the used unit for speed commands.

TABLE 7: HANDOVER COMMAND EXAMPLE

Utterance	portugal one lima whiskey tower one one eight one five
Transcription	TAP1LW CONTACT TOWER TAP1LW CONTACT_FREQUENCY 118.150
Explanation	The frequency always consists of six digits, i.e. ending zeros are not ommitted even if they are not spoken. Splitting the handover into two commands (CONTACT of position and CONTACT_FREQUENCY with frequency value) might seem odd, but otherwise we would have a command with two logically different values. With this approach we can also transcribe situations in which one part is missing as in “portugal one lima whiskey to tower”. We also split QNH and DESCEND into two parts.

TABLE 8: REPORT AND REQUEST COMMAND EXAMPLE

Utterance	tunair five one four report your speed
Transcription	TAR514 REPORT SPEED
Explanation	REPORT is the command type and SPEED is the qualifier. We did not decide for a command REPORT_SPEED (with underscore) to avoid inventing two many different command types. In the same way we decided for a pilot request, e.g. TAR514 REQUEST PUSH_BACK.

TABLE 9: TAXI CLEARANCE FROM TOWER/GROUND CONTROLLER EXAMPLE

Utterance	swiss three five four hold short of runway two eight left
Transcription	SWR354 HOLD_SHORT RW28L
Explanation	We add the string “RW” to each runway transcription. We also add the letters L (left), R (right), C(center) to a runway clearance if the airport has more than one runway with that direction. If no runway is said, ‘none’ is mandatory.

E. Conditional Clearances

A conditional clearance issued by an ATCo, only becomes effective depending on the satisfaction of a requirement [26]. The conditional clearance contains one of the following keywords: UNTIL, WHEN, or IF (see Fig. 11). The syntax of the “Condition” part in Fig. 11 is detailed in the following Fig. 12 and 13.

Most keywords of the condition (e.g. PASSING) can be used both with UNTIL and with WHEN. Different to PASSING, we use REACHING if the state remains for some time. The “Command” part in Fig. 13 is the same as the “Command” part in Fig. 6.

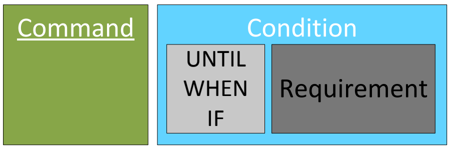


Figure 11. Clearance, with Command, Conjunction and Requirement.

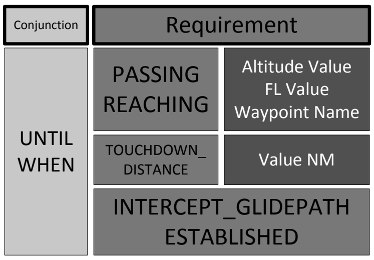


Figure 12. Elements of a condition (part 1).

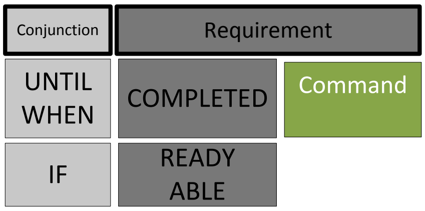


Figure 13. Elements of a condition (part 2).

Again, we have some examples to explain this.. Conditional parts in Table 10 to Table 14 are highlighted in green.

TABLE 10: CONDITIONAL CLEARANCE WITH UNTIL EXAMPLE

Utterance	finnair five kilo alpha descend with fifteen hundred feet per minute or greater until passing flight level seven zero
Transcription	FIN5KA RATE_OF_DESCENT 1500 ft_min OR_GREATER UNTIL PASSING ALTITUDE 70 FL
Explanation	The unit for vertical speed commands is “ft_min” or “none”, if not specified in clearance. OR_GREATER, OR_LESS, GREATER and LESS are qualifiers for vertical and horizontal speed clearances. The vertical speed clearance starts now and ends when flight level 70 is reached. Therefore UNTIL is used.

TABLE 11: CONDITIONAL CLEARANCE WITH WHEN EXAMPLE

Utterance	lufthansa four delta when you pass flight level six five cleared direct mike india charlie
Transcription	DLH4D DIRECT_TO MIC WHEN PASSING ALTITUDE 65 FL
Explanation	The condition is in the middle of the clearance, but in command transcription the condition is always added (according to Fig. 11) at the end.

TABLE 12: CONDITIONAL CLEARANCE WITH IF EXAMPLE

Utterance	b_air seven one two if able turn left heading two seven zero to intercept localizer runway two four
Transcription	ABP712 HEADING 270 LEFT IF ABLE
Explanation	The command did not contain a clearance to intercept the localizer. This will depend on ability of ABP712 to turn left on heading 270. It is just an information that the turn is for future interception of localizer of runway 24.

TABLE 13: COMPLEX CONDITIONAL CLEARANCE WITH DIFFERENT CONDITIONS

Utterance	csa seven zero seven hotel when established localizer reduce speed one six zero knots to maintain until four miles final
Transcription	CSA707H REDUCE 160 kt WHEN ESTABLIHED LOC UNTIL TOUCHDOWN_DISTANCE 4 NM
Explanation	"csa" is the airline indicator, written in one string. We first have a WHEN-part. Reduction to 160 knots is not started now, but later, when aircraft is established on localizer. New speed is not kept forever (until next command), but only until distance to touchdown is four miles (outer marker passed).

The second condition in Table 13 could also be transcribed as a new command, i.e., "CSA707H MAINTAIN_SPEED 160 kt UNTIL TOUCHDOWN_DISTANCE 4 NM," because the reduction is not performed until 4 miles final, and the speed is kept. However, "reduce speed one six zero knots until four miles final," is also transcribed as "REDUCE 160 kt UNTIL …" and not as "MAINTAIN_SPEED 160 kt UNTIL".

TABLE 14: COMPLEX CONDITIONAL CLEARANCE EXAMPLE

Utterance	csa five four three if able proceed to final approach fix runway two four and descend four thousand feet qnh one zero two three
Transcription	CSA543 DIRECT_TO FAF24 IF ABLE CSA543 DESCEND 4000 ft IF ABLE CSA543 INFORMATION QNH 1023
Explanation	FAF24 is assumed as the data base name for the final approach fix. It is assumed that the DESCEND command is also conditional, but the QNH value is valid without condition.

The examples above show that conditional clearances are complex commands. They are used by controllers regularly as they reduce communication workload by reducing number of transmissions. Improvements of ontology with respect to conditional clearances are expected in the near future.

IV. CONTROLLER PILOT DATA LINK COMMUNICATIONS

Controller Pilot Data Link Communications (CPDLC) is a method by which air traffic controllers can communicate with pilots over a datalink system [25]. Although mandatory for aircraft entering oceanic airspace (e.g. Shanwick Oceanic control), it is not mandatory for the majority of short haul aircraft. As such, its deployment and use on small aircraft is incredibly limited. At NATS e.g., CPDLC may only be used for oceanic messages and in limited other circumstances domestically. Thus, voice remains the primary and official means of communication between controllers and pilots. Although more and more aircraft are utilizing CPDLC, the deployment of it is still limited to the ANSPs offering the service. Guaranteed transmission times of CPDLC are longer than those of voice [27]. Therefore, Global Operational Data Link (GOLD) Manual [28] states "To minimize pilot head down time and potential distractions during critical phases of flight, the controller should use voice to communicate with aircraft operating below 10,000 ft AGL" (sect. 4.1.2.3) and "If a conflicting CPDLC and voice communication is received, the controller should obtain clarification using voice" (sect. 4.1.2.6).

CPDLC has been demonstrated as offering a good 'monitoring' and 'information' service concerning certain orders and commands from ATC to pilots. However, CPDLC is a limited sub-population of ICAO phraseology. Nevertheless, CPDLC is a standard to formalize command and information exchange between controllers and pilots and vice versa. For exchange of enroute clearances the ontology defined by CPDLC is very powerful. PJ.16-04 ontology on the other hand also covers ground and tower controller phraseology (e.g. command types LINEUP, FOLLOW, TAXI, VACATE detailed in Fig. 10 are not covered at all by CPDLC).

Landing clearances (e.g. CLEARED ILS, CLEARD NDB) are not covered by CPDLC. Also, no special command type exists for maintaining a heading value, although the CPDLC types UM190 FLY HEADING [degrees] resp. UM94 TURN [direction] HEADING [degrees] could be used. The semantics for the aircraft/pilot would be the same, but our aim is to transcribe what the controller has said. There is a difference between a maintain heading and just a heading clearance. The value of a maintain-heading clearance should be near the value of the current aircraft heading. INFORMATION command type for QNH or wind direction information exchange is also not foreseen.

On the other hand PJ.16-04 phraseology does not cover CROSS command type for crossing flight levels or waypoints (e.g. UM47 CROSS [position] AT OR ABOVE [leve]). CPDLC also considers which command type may be combined in a transmission and which may not. In this way CPDLC also models conditional clearances. PJ.16-04's intention, however, is to model what the controller has said independent of whether it is allowed or even makes sense.

V. APPLICATION OF ONTOLOGY

The ontology described in chapter III is developed within SESAR solution PJ.16-04. The five different ASR exercises of PJ 16-04 are detailed in this chapter with respect to first proof-of-concept cases of the developed ontology.

A. *ASR increasing Controller Safety at Munich Approach*

DFS is already using Voice Recognition and Response (VRR) very extensively in the training environment as simulation pilot replacement. As part of SESAR solution PJ.16-04, DFS wants to go to the next step and make ASR available at the controller's side. The task is to analyze in how ASR would increase safety in the current work of the Munich approach unit. The following areas are identified:

- Target Location Assistance (TLA): The callsign in each controller clearance shall be recognized and highlighted on the controller air situation window, so that the controller can easily identify the targets' location.
- Controller Clearance Verification (CCV): With this step, the spoken controller clearance will be checked against the manual system input of the controller. In case of a deviation, a warning will be shown.

Both activities will improve the existing ATC safety net. Hours of training simulation recordings and recordings of spoken Navigational Reference Points were needed in the past to create and improve a German accented English voice model. As this process is still ongoing, a standardized ontology will ease the exchange of voice recordings and further transcriptions.

B. *Safety assessment of ASR for radar label maintenance*

DLR and Saarland University have shown in the AcListant® project that command recognition rates of 95% with command recognition error rates below 2% are possible [29]. Its follow-up project, AcListant®-Strips validated that Assistant Based Speech Recognition (ABSR) can reduce controllers' workload for radar label maintenance by a factor of three [30] and that fuel savings of 60 liters of kerosene per flight are possible [11]. In this exercise Air Navigation Services of Czech Republic (ANS CR), DLR, the aviation consultant Integra and Thales ATM Group concentrated on the safety aspects of ASR when used as input device for radar label maintenance instead of a mouse. From a safety perspective, a low command recognition error rate combined with a high command recognition rate is naturally desirable. However, in addition to detecting any/all errors it is important to allow the controller to identify errors where and when they may occur to be able to override the ASR derived data and to take appropriate corrective action through manual inputs. The importance of more or less constant recognition rate over the instruction variety was identified as well. Therefore, the ASR performance needs to be known by the ATCo. The roadmap for integration of an ASR application into an operational CWP is addressed. The Nuance speech recognizer is used as the engine supported by DLR's Arrival Manager 4D-CARMA for predicting the set of possible controller commands in order to reduce the search lattice of the ASR engine.

The derived ontology is used for transcription of training data respectively for transformation of existing training data. It is also needed to define the interface between the command prediction module (the assistant system) and the speech recognition engine on the one hand and between the speech recognition engine and the controllers' HMI on the other hand.

C. *Calculation of Controller's Workload*

In this use case, ENAIRE and CRIDA will apply ASR to Enroute and E-TMA (extended terminal manoeuvring area) environment. The application of ASR (i.e. VOICE [31]) for supporting calculation of controller´s workload will be analyzed. During the exercise, CRIDA will use VOICE ASR to automatically transcribe voice recordings and determine the associated ATC event (abstraction of command types in ontology). Then the events will be used for controller workload estimation [6], [7].

The quality of the measured workload is evaluated afterwards against the workload calculated with the true events and against the given workload by controllers. The exercise will take place in one of the validation activities of SESAR2020 PJ.10-01b "Low Complexity Environment", focused on Madrid Area Control Center.

D. *ASR Application in a Remote Tower Environment*

In this exercise, DLR and HungaroControl concentrate on ASR application in a Multiple Remote Tower Environment. Simulation airports are the Hungarian airports; Budapest, Debrecen, and Pápa. No speech recognition system is used. However, DLR will develop a Command Hypotheses Predictor continuously predicting the set of controller commands which are possible in the current traffic situation. The quality of the predicted command set will be evaluated against actually given commands. If prediction accuracy is acceptable, it demonstrates that the AcListant® approach of ABSR [29] could be transferred from the approach domain to the tower domain. Again, command prediction and transcription of given commands need the derived ontology.

E. *ASR Supporting Situational Awareness in a Multi-Remote Tower Environment*

Frequentis together with DFS are developing a multi remote tower concept, which is validated for different German airports. To improve the situational awareness of the controller, operating multiple remote towers, it is foreseen to investigate the use of ASR. Especially for situations with parallel operations, it is important to have a complete overview of the ongoing communications (controller and pilot) of each airport and to distinguish between each other. To support the operator in the best possible way, a special focus is set on integrating the recognized information in a user-friendly way within the HMI. To provide the best results between the human and the machine it is very important to extract the relevant information from the actual voice transmission. This task will be eased by the developed ontology.

F. *Ontology and Stardardization*

Fig. 14 summarizes the five experiments with respect to used input and output interfaces and involved modules. Green arrows show interfaces when benefiting from described ontology, i.e. when abstraction of recognized words resp. predicted commands are needed. As firstly described in [29] ABSR relies on a module (Command Predictor in Fig [14]), which periodically predicts a set of possible commands with respect to the current air traffic situation, i.e. with respect to radar data etc. These predicted commands are the input of the speech recognizer to reduce its search space.

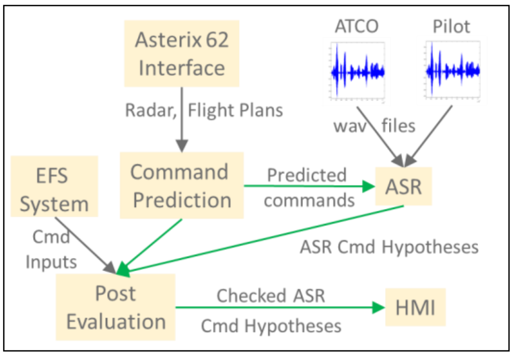


Figure 14. Module of PJ.16-04 experiments and its in- and outputs

TABLE 15: EXERCISES, INTERFACES AND MODULES

Interface or Module Name	*Ex. 1 (V.A)*	Ex. 2 (V.B)	*Ex. 3 (V.C)*	Ex. 4 (V.D)	Ex. 5 (V.E)
Radar, Flight Plan		used		used	
Cmd Inputs	used				
Wav files		used	used		used
Predicted Cmds		used		used	
ASR Cmd Hypotheses	used	used	used		used *
Checked ASR Cmd Hypotheses	used	used	used *		used
Asterix 62 Interface		used		used	
EFS System	used				
ASR (Pilot)					used
ASR (ATCo)	used	used	used		used
Command Prediction		used		used	
Post Evaluation	used	used	used *		used *
HMI	used	used	used *		used

Ex = Exercise, Cmd = Command, in darker grey we put the modules, in lighter grey the interfaces; we add a star (*, if more information is provided in the text.

Unfortunately the output of ASR are not always the correct spoken commands (with respect to ontology), but only a best guess of ASR, i.e. a set of command hypothesis or a set of sets of command hypotheses with plausibility values. The controller, however, is not interested that his/her spoken command could be “REDUCE 160 kt” with a plausibility value of 0.9 or “HEADING 160 LEFT” with a plausibility value of 0.6. He/she prefers one unique output and in the best case this is the spoken one. Therefore, the Post Evaluation module is needed. It uses different input sources (e.g. mode-S, set of predicted commands or controller’s input into the Electronic Flight System, EFS) to reduce the different command hypotheses sets to a unique command hypotheses. This is the input for the HMI, i.e. the controller. Not all modules and not all interfaces are needed in each of the five exercises. Table 15 shows which of the interfaces are implemented for each of five exercises. The exercise of CRIDA and ENAIRE also uses an HMI (sect. V.D), but here the HMI does not show the checked output to the controller, but uses this output to estimate the controller’s workload. In exercise 3 and 5 it is currently undecided if a Post Evaluation of the ASR will be implemented. If not, the output of the ASR system is directly used by the HMI / Workload Prediction.

VI. CONCLUSIONS AND NEXT STEPS

Previously PJ.16-04 partners have developed ASR systems independently or in groups with a small number of partners involved. In SESAR PJ.16-04 solution, however, main European suppliers from ATM industry, Air Navigation Service Providers, and ATM research are working together and are uniting their knowledge of speech recognition technology. They have agreed on a common ontology for controller command transcription. DLR, Austro Control, and ANS CR have transcribed more than 9,000 controller commands for the MALORCA project [22]. DLR has transcribed more than 11,000 controller commands for the AcListant® strips validation trials [11], [30].

The commands from our ontology cover approach, enroute, tower, and ground control. By introducing type qualifiers (e.g. LESS, OR_GREATER) and additional type strings (e.g. ILS, NDB), the number of different command types was reduced to approximately 100, resulting in 14 different command type categories. Additionally units have been agreed. Most challenging was the discussion and solution for the conditional clearances. The authors assume that approximately 95% of the commands from partners’ command data bases will be covered now by the developed ontology at least for the approach domain. This will enable the discussed applications, i.e., read back cross checking, easy exchange of transcription between different stakeholders, interfacing between controller between controller HMI and ASR system, cross checking with Mode S output or with flight label contents. Controller workload prediction is eased because different command types result in different challenges with respect to controller workload (an ATIS information is different from a clearance with DIRECT_TO). On the other hand different command types could now be clustered very easily (e.g. CLIMB, DESCEND, ALTITUDE). Furthermore, a safety analysis is enabled when checking transcriptions of real world controller utterances against ICAO standard phraseology. Collectively, these suggest real tangible benefits from the ontology beyond command transcription.

As a next step it will be proven that the agreed ontology can also be applied to future controller and pilot utterances. Five exercises with respect to ASR, described in chapter V, are conducted within SESAR 2020. They are coordinated within PJ.16-04 solution. With the acknowledgement that 100% will never be covered, lessons learnt from these verification exercises will help to further improve the presented ontology and to reduce the remaining 5% of the controller utterances in an attempt to get as close as possible to a complete system. Integration with CPDLC and standardization, not limited to the PJ.16-04 partners or Europe, should be the next logical step. PJ.16-04 hopes to have initiated this process with this contribution.

The exercises within SESAR2020 will clarify the implement ability of ASR for ATC operations. It will also improve the understanding of the impact of ASR on controller performance. This in turn will provide a useful input to the development of future concepts of operations, improve training and procedures and aid in the designing of future supporting tools.

ACKNOWLEDGMENT

We would like to thank the partners of MALORCA project (Saarland University, Austro Control and Idiap Research Institute) and the controllers and pilots who anonymously provided us with real world command examples.

REFERENCES

[1] C. Hamel, D. Kotick, and M. Layton, "Microcomputer system integration for Air Control Training," Special Report SR89-01, Naval Training Systems Center, Orlando, FL, USA, 1989.

[2] FAA, "2012 National aviation research plan (NARP)," March 2012.

[3] D. Schäfer, "Context-sensitive speech recognition in the air traffic control simulation," Eurocontrol EEC Note No. 02/2001 and PhD Thesis of the University of Armed Forces, Munich, 2001.

[4] R. Tarakan, K. Baldwin, and R. Rozen, "An automated simulation pilot capability to support advanced air traffic controller training," 26th Congress of the International Council of the Aeronautical Sciences, Anchorage, AK, USA, 2008.

[5] S. Ciupka, "Siris big sister captures DFS,“ original German title: "Siris große Schwester erobert die DFS," transmission, Vol. 1, 2012.

[6] J.M. Cordero, M. Dorado, and J.M. de Pablo, "Automated speech recognition in ATC environment," Proceedings of the 2nd International Conference on Application and Theory of Automation in Command and Control Systems (ATACCS'12), IRIT Press, Toulouse, France, 2012, pp. 46–53.

[7] J.M. Cordero, N. Rodríguez, J.M. de Pablo, and M. Dorado, "Automated speech recognition in controller communications applied to workload measurement," 3rd SESAR Innovation Days, Stockholm, Sweden, 2013.

[8] S. Chen, H.D. Kopald, A. Elessawy, Z. Levonian, and R.M. Tarakan, "Speech inputs to surface safety logic systems," IEEE/AIAA 34th Digital Avionics Systems Conference (DASC), Prague, Czech Republic, 2015.

[9] S. Chen, H.D. Kopald, R. Chong, Y. Wei, and Z. Levonian, "Read back error detection using automatic speech recognition," 12th USA/ Europe Air Traffic Management Research and Development Seminar (ATM2017), Seattle, WA, USA, 2017.

[10] M. Kleinert, H. Helmke, G. Siol, H. Ehr, M. Finke, A. Srinivasamurthy, "Machine learning of controller command prediction models from recorded radar data and controller speech utterances," 7th SESAR Innovation Days, Belgrade, 2017.

[11] H. Helmke, O. Ohneiser, J. Buxbaum, and Chr. Kern, "Increasing ATM efficiency with assistant-based speech recognition," 12th USA/Europe Air Traffic Management Research and Development Seminar (ATM2017). Seattle, Washington, 2017.

[12] Z. Tüske, K. Irie, R. Schlüter, and H. Ney, "Investigation on log-linear interpolation of multi-domain neural network language model," in Proc. of IEEE Int. Conf. on Acoustics, Speech, and Signal Processing, Shanghai, China, 2016, pp. 6005–6009.

[13] W. Xiong, J. Droppo, X. Huang, F. Seide, M. Seltzer, A. Stolcke, D. Yu and G. Zweig, "Achieving human parity in conversational speech recognition,", Technical Report MSR-TR-2016-71 Microsoft Research, arXiv preprint arXiv:1610.05256 2017.

[14] S. F. Chen, B. Kingsbury, L. Mangu, D. Povey, G. Saon, H. Soltau, and G. Zweig, "Advances in speech transcription at IBM - under the DARPA EARS program," in IEEE Transactions on Audio, Speech and Language Processing, Vol. 14, No. 5, Sep. 2006, pp. 1596-2608

[15] V.N. Nguyen and H. Holone, "N-best list re-ranking using syntactic score: A solution for improving speech recognition accuracy in Air Traffic Control," 16th Int. Conf. on Control, Automation and Systems (ICCAS 2016), Gyeongju, Korea, 2016, pp. 1309–1314.

[16] V.N. Nguyen and H. Holone, "N-best list re-ranking using syntactic relatedness and syntactic score: An approach for improving speech recognition accuracy in Air Traffic Control," 16th Int. Conf. on Control, Automation and Systems (ICCAS 2016), Gyeongju, Korea, 2016, pp. 1315–1319.

[17] D.R. Johnson, V.I. Nenov, and G. Espinoza, "Automatic speech semantic recognition and verification in Air Traffic Control," IEEE/AIAA, 32rd Digital Avionics Systems Conference (DASC), East Syracuse, NY, USA, 2016

[18] http://www.json.org, JSON = JavaScript Object Notation, n.d.

[19] AcListant homepage: www.AcListant.de, AcListant = Active Listening Assistant, n.d.

[20] A. Schmidt, "Integrating situational context information into an online ASR system for Air Traffic Control," Master Thesis, Saarland University (UdS), 2014.

[21] Y. Oualil, M. Schulder, H. Helmke, A. Schmidt, and D. Klakow, "Real-time integration of dynamic context information for improving automatic speech recognition," Interspeech, Dresden, Germany, 2015.

[22] MALORCA homepage: www.malorca-project.de, MALORCA = Machine Learning of Recognition Models for Controller Assistance, n.d.

[23] A. Srinivasamurthy, P. Motlicek, I. Himawan, G. Szaszák, Y. Oualil, and H. Helmke, "Semisupervised learning with semantic knowledge extraction for improved speech recognition in air traffic control," INTERSPEECH 2017, 18th Annual Conference of the International Speech Communication Association, Stockholm, Sweden Aug. 2017.

[24] D. Randall, "Direct Voice Input (DVI) Technology readiness and status introduction," Whitely, Fareham, UK, 2006.

[25] Eurocontrol: "LINK2000+: ATC data link operational guidance in support of DLS regulation," No 29/2009, Vol.6. 17. December 2012, online available at https://www.skybrary.aero/bookshelf/books/2383.pdf

[26] https://www.skybrary.aero/index.php/Conditional_Clearance, n.d.

[27] O. Veronika Prinzo: "Data-linked pilot reply time on controller workload and communication in a simulated terminal option," Civil Aeromedical Institute, Federal Aviation Administration, Oklahoma City, Oklahoma 73125, May 2001.

[28] ICAO: "Global operational data link document (GOLD)," 2nd Edition, 2013-04-26.

[29] H. Helmke, J. Rataj, T. Mühlhausen, O. Ohneiser, H. Ehr, M. Kleinert, Y. Oualil, and M. Schulder, "Assistant-based speech recognition for ATM applications," 11th USA/ Europe Air Traffic Management Research and Development Seminar (ATM2015), Lisbon, Portugal, 2015.

[30] H. Helmke, O. Ohneiser, T. Mühlhausen, and M. Wies, "Reducing controller workload with automatic speech recognition," IEEE/AIAA 35th Digital Avionics Systems Conference (DASC), Sacramento, California, USA, 2016.

[31] N. Rodríguez and J.M. Cordero, "Relationship between workload and duration of ATC voice communications," 6th International Conference on Research in Air Transportation, Istanbul, 2014.