The Integrated Diagnostic System (IDS): Remote Monitoring and Decision Support for Commercial Aircraft - *Putting Theory into Practice*

Michael Halasz, François Dubé, Bob Orchard
Integrated Reasoning Group
National Research Council of Canada
Institute for Information Technology
M50 Montreal Road
Ottawa, Ontario, Canada K1A 0R6
[mike.halasz, francois.dube, bob.orchard]@iit.nrc.ca

Robert Ferland
Program Leader, Decision Support Systems
Air Canada
PO Box 9000
Saint Laurent, Quebec, H4Y 1C2
rferland@aircanada.ca

Abstract

The Integrated Diagnostic System (IDS) is an applied artificial intelligence (AI) project that deals with remotely monitoring a fleet of commercial aircraft and proactively alerting maintenance staff to problems which could disrupt operations. IDS uses a number of AI techniques to analyse situations and provide recommendations. IDS is very much a work in progress and the research has been adapted to the needs of an airline operator. Due to the ongoing working relationship with Air Canada, the system deals with many airline integration issues such as network communications, document delivery, technology transfer, and equipment dispatch regulations. The objective of this paper is to share some of the insights gained over the course of this project and to highlight practical issues of introducing advanced information technology systems into an airline's environment.

Introduction

In 1992, researchers at the National Research Council of Canada (NRC) initiated the Integrated Diagnostic System (IDS) program and solicited the participation of industrial partners. The goals of this effort were to develop a platform to *research*, *develop*, *integrate*, and *test* advanced diagnostic and decision support tools for the maintenance of complex equipment. The partners would use this knowledge to enhance existing products, develop new products, improve operations, and establish new research directions. It was felt that the effort would have benefits for maintenance operations in many application areas including aerospace, power generation, manufacturing, forestry, and transportation.

There were a number of reasons for choosing the maintenance domain that continue to be valid[1]. For example:

- Improved interpretation of an organization's diverse and distributed information assets can help the maintenance mission. Novel AI solutions could be developed and tested in real life situations.
- There is a need for systems which help in *time critical, accurate,* and *consistent* decision making.
- Staff are progressively required to do *more with less* with continual pressure to *maximize equipment utilization.*
- Technicians deal with far more *equipment diversity* than ever before.
- Access to the *right data, knowledge* and *expertise* at the *right time* is necessary to effectively carry out the appropriate repair action.
- The increasing amounts of useful data provided by newer generation equipment require tools which aid in *integration* and *interpretation.*
- *Unscheduled ("surprise") maintenance* is persistent.
- Significant savings can be realized through reduction in *no fault found* and *repeat snag*[2] *handling.*
- Innovative information technology solutions can improve the bottom line. Conservative estimates of savings for the airline and mining industries are 2-4% and 5-8% of the total maintenance budget, respectively[3].

A high level indication of the 1996 costs of maintenance across 11 Canadian industry sectors is shown in Figure 1. It indicates that, in addition to every dollar spent on new machinery, an additional 58 cents is spent on maintaining existing equipment. This amounts to repair costs of approximately $15 billion per year. Comparing this with 1990 data, the ratio of money spent on maintenance has risen by about 14%. It is also worth noting that for some sectors such as railways and truck transport, maintenance exceeds the cost of new machinery.

IDS Requirements

To develop and test envisioned concepts, a good "laboratory" having the following key attributes was needed:

- availability of data,
- free access to systems and personnel, and
- a complex, distributed operation with significant impact of downtime.

1. Recent industry reports corroborate this position (e.g. [Canaan Group 1998]).

2. An observed operational anomaly. Typically referred to as a "squawk" in USA.
3. Based on NRC studies which attempted to quantify the economic benefits of assessing equipment problems correctly.

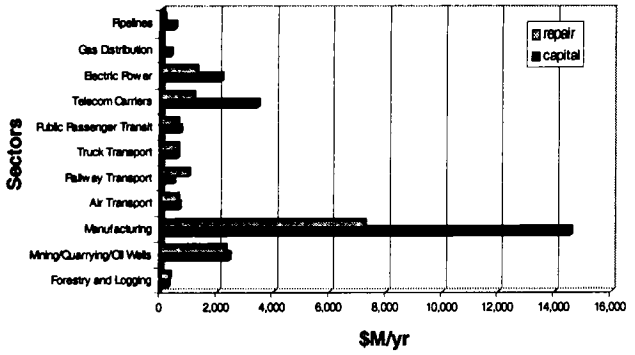

Figure 1. repair vs. capital expenditures
(Source: Statistics Canada,1996)

Large commercial airlines fit the bill very nicely. In the fall of 1993, Air Canada became a participant in the project and a four month requirements analysis was carried out.

This study was a top-level requirements analysis. It concentrated on developing a broad understanding of the equipment operator's maintenance operations and the role systems such as IDS might play within the organization. Quantifying the benefits from effective use of available data to improve diagnostic performance proved to be quite challenging. Nevertheless, a methodology was developed which attempted to measure a subset of direct benefits in order to show sufficient promise for the idea. Thus, there is justification in assuming that the benefits of such systems, effectively deployed, are much larger than indicated in the study.

The main highlights of the study were:

- IDS would cut across many different departments in the organization that did not typically interact.
- Any solution had to deal with *all* equipment systems and *all* fleet types. This perspective is at odds with most manufacturers since their goal has traditionally been to become sole source suppliers of equipment and associated support tools - a view not shared by their customers.
- Decision making is highly distributed. Timely access to the right data, knowledge, and expertise is necessary to effectively carry out the maintenance mission. Advice from the system had to be presented in real-time to a diversity of users.
- Newer generation equipment produces increasing amounts of potentially useful data. These data are not optimally utilized due to large volumes and difficult to detect patterns. Innovative information technology was required to aid in the integration of interpretation of data.
- Ideally, systems such as IDS, should have *prognostic* as well as "*repair management*" capabilities to be truly successful. Repair management provides the ability to recommend the appropriate course of action given the broader context within which a problem occurs. Factors such as, schedules, location, weather, and passenger traffic would be taken into account. A very simple example would be to automatically order a part and deliver it to an airport before the aircraft lands in order to avoid a major disruption.

The addition of these two sets of functions is expected to generate further tangible benefits which could be much more significant than the diagnostic benefits although perhaps more difficult to achieve technically.

In parallel with the requirements studies, an airline market assessment indicated conservative savings of at least $1.5B per year for the world's airlines with more than 40 aircraft in their fleet (for improved diagnostics alone).

Air Canada Operational Environment

Air Canada currently has a fleet of eight different aircraft types from four airframe manufacturers (160 aircraft). This is typical of a medium sized commercial carrier.

Figure 2 depicts the world within which a system such as IDS should operate. Aircraft are continually on the move around the globe and turn-times at the gate continue to be shortened to maximize utilization. This creates many challenges. A number of functional groups within the airline can be involved in maintenance decisions depending upon the problem. They are briefly described below:

- *Line technicians* repair aircraft at the gate or on overnight layover. The prime objective is to safely turn aircraft around with minimal disruption.
- The *Maintenance Control Centre* views the entire fleet and thus obtain a broad perspective. They deal with problems reported by the pilot or on-board systems. They also monitor fleet status, identify trends, deal with persistent/ foreseeable problems, and decide maintenance policy.
- *Engineering* looks at specific performance indicators of the equipment and will only become involved with difficult immediate concerns, on an as-required basis. They typically have the longest decision horizon.
- The *manufacturers representative* can get involved in certain difficult problems.
- The personnel in *parts stores* must ensure that an adequate supply of spares exists from the various production sources both within and external to the airline.
- The goal of *System Operations Control* is to keep the entire fleet flying on schedule. They make system wide decisions on factors such as, disruptions due to weather or equipment failure, and flight crew readiness.

Modern aircraft have on-board systems which can transmit data to ground stations.[1] These data consist of routine performance snapshots (engine parameters, pressures, altitude, valve positions, temperature...), typed pilot messages, aircraft generated fault codes, and limit exceedance reports.

Many databases support maintenance. They contain descriptions of symptoms and associated maintenance actions (free form text), deferred problems, flight schedules (static and dynamic), weather, component reliability, check schedules, and parts location. There is also a wealth of useful information held at the manufacturer, and by people and information systems in the engineering and maintenance control departments. This is not widely distributed and thus not available to the line technician in a timely manner.

When aircraft problems occur, technicians rely heavily on professional judgement and knowledge. Additional support

1. A particularly interesting aspect of Air Canada is the datalink system which transmits aircraft generated messages to a ground based database. This permitted development of a system that would constantly monitor and interpret these data to allow for actions to take place before the arrival of the aircraft at its destination.

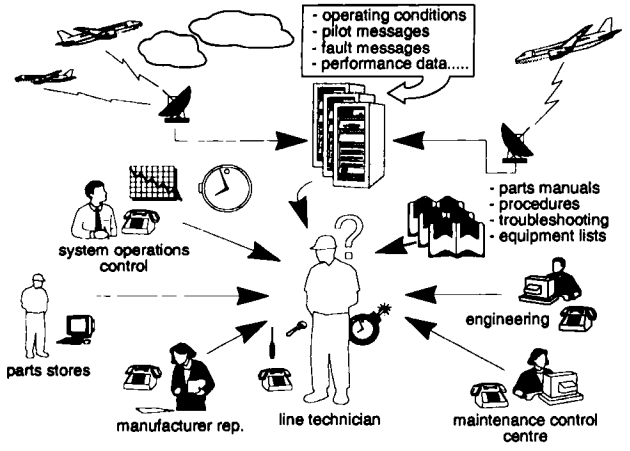

Figure 2. Current situation - airline

tools at their disposal are the:

- aircraft Built-In Test Equipment (BITE),
- Minimum Equipment List (MEL) - a document governing minimum equipment needs to dispatch aircraft safely,
- aircraft log book which provides free form text descriptions of the problems encountered,
- post flight reports (a summary of the messages generated by the aircraft for a given flight leg), and
- troubleshooting manuals (migrating to electronic media)

Thus, the decision environment is characterized by distributed data, information, and knowledge sources. Making timely and correct decisions is important and requires access to various combinations of these sources.

Approach

An overriding consideration in conceiving IDS was that it had to address *all* aircraft types. With few exceptions (i.e. where fleet economy of scale permits), this is the reality which most line technicians face. A different system for each aircraft type was not going to be accepted and would not be practical from a deployment point of view. This initial assumption set the stage and precluded AI techniques such as in-depth model-based reasoning or large knowledge bases of specific rules for certain aircraft types.

The philosophy was very much top down. In other words, a lot of time was spent with technicians analysing how they reasoned and exploiting inherent structures in the troubleshooting manuals to generate rule sets automatically. Thus, as a first cut at the problem, this approach elicited methodologies pertaining to the investigative process and should extend to other aircraft types.

Once the high level functionality of IDS was defined, important decisions were made concerning the development environment and associated tools. Concurrent with this was an intensive knowledge acquisition process. It's objectives were to obtain an in-depth understanding of:

- business processes associated with Air Canada's aircraft maintenance organization (Technical Operations),
- data (including database structures), information, and knowledge sources,
- interrelation and potential usefulness of the different information systems, and
- specific diagnostic rules of thumb.

All along it was evident that organizations such as Air Canada are in a constant state of flux, thus an evolutionary protoyping methodology was followed [see, for example, McConnell 1996]. The process involved frequent feedback from users and an evolving set of requirements.

Functional Architecture

Figure 3 provides an overview of the IDS architecture. Not all of this exists since it incorporates the broader vision.

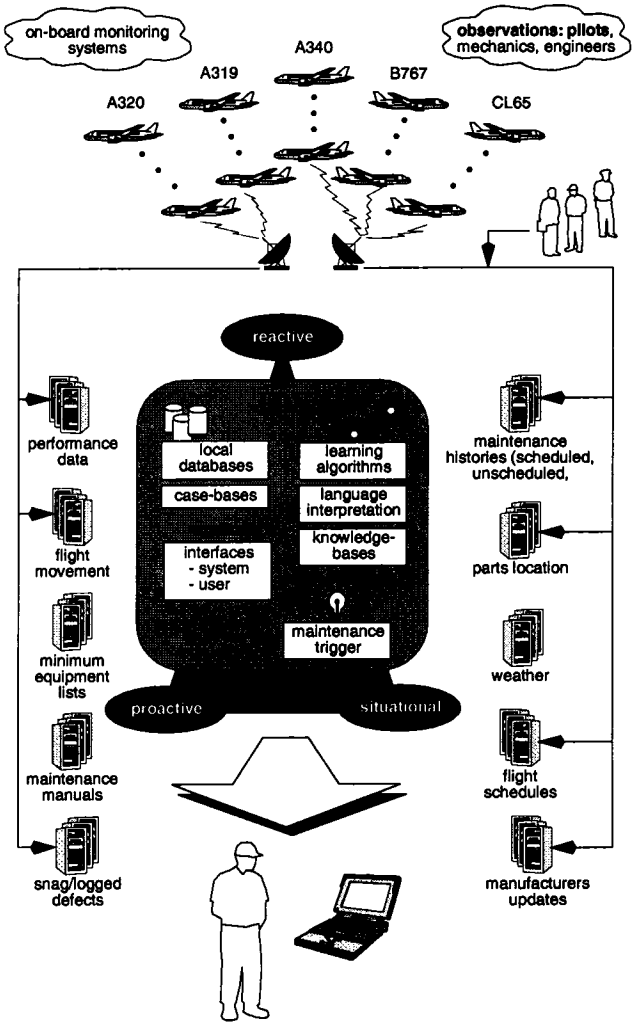

Figure 3. IDS functional architecture

IDS is data driven. The data driving it is stored in a multitude of dispersed databases of various vintages within Air Canada. Pertinent data are "pushed" to IDS and written to a database. These data coupled with the interpretive logic within IDS then triggers certain maintenance actions. Once alerts are generated by IDS, it continues its "investigation" by requesting subsets of data to refine its recommendations.

In order to be truly effective IDS must possess a number of basic functionalities. First of all, it must have reliable inter-

faces to various distributed databases and sources of electronic manuals. Secondly, it must maintain its own internal data sources and case-bases for reasoning. In addition, it incorporates AI techniques to interpret text strings and rules, learn patterns, reason about situations, and build and retrieve cases. Finally, it must address the delivery of information to a global and mobile work force.

The ultimate vision for IDS includes a:

- *reactive* component which efficiently helps find the root cause of a problem that happened without warning,
- *proactive* component which deals with problems in which some sort of "trend" or pattern is evolving, and
- *situational* component which takes into account the larger context of a problem whether it be reactive or proactive.

Basic Operating Principle

Every time a message is received, IDS determines, whether or not the message belongs to an ongoing problem, is the start of something new, or can be ignored. The process exploits many knowledge sources, some allowing messages to cluster, others allowing messages (or clusters) to merge, be modified, or discarded. The ideal result is clear, concise, and complete descriptions of fault events associating symptoms and correct repair actions. Once validated, these associations are added to a case database for future retrieval. Ultimately, this can lead to automatic case creation - seen as being highly useful by airline personnel.

In addition, for a given symptom cluster, IDS determines the impact on aircraft dispatch, and has built-in links to the appropriate pages in the troubleshooting manuals and the minimum equipment list (MEL).

A Simple Example

Currently, IDS uses two asynchronously generated data sources. One is aircraft generated *performance and fault data*. The other is *snag/logged defect* information entered by technicians and flight crews. The latter is very similar to the free form text entered at car dealers when describing problems and associated repairs.

An example of a sequence of events for a particular flight is shown in Table 1. For illustration purposes, only the essence of the messages are given. At 13:04:12 the aircraft takes off. Over the course of the 2 hour flight 22 messages were processed by IDS. All were generated by on-board systems except for two messages (13:22:41 and 14:47) which were entered by the pilot and flight crew respectively. The last message occurred after the flight and was entered into the system at 22:52. It describes the corrective action.

This example shows the snag was first picked up at 13:09:14. At this point IDS alerted ground crews to a possible dispatch problem upon arrival. At 13:10:08 an associated message was generated and added to form a symptom cluster. At 13:22:41 the pilot confirms the situation and IDS adds this message to the previous two and assigns a higher alarm level to the snag.

While all this is going on, IDS does three more things. As the case dynamically evolves, it will, when requested:

- retrieve closest neighbours (previously validated cases) and present them to the user,
- recall related historical problems for the particular aircraft, and
- determine the logical entry point in the electronic manuals and MEL for instructions on problem relief.

TABLE 1. Sample flight leg data

Time	Phase	Event	Description
13:04:12	Take off	NAV	
13:08:34	Cruise	WRN	ENG 2 OVSPD PROT FAULT
13:08:35	Cruise	FLR	HMU (OSG), J7 // EIU2FAD
13:09:14	**Cruise**	**WRN**	**ENG 2 COMPRESSOR VANE**
13:10:08	**Cruise**	**FLR**	**VBV ACT, HMU ENG2A // EIU2FADEC**
13:19:50	Cruise	WRN	SFCS
13:20:56	Cruise	FLR	FLP LH PROX SNSR 1 37 CVOR LGCIU 1 // SFCC 1
13:22:28	Cruise	FLR	L FLP DISC PROX SNSR 37CV // LGCIU 1
13:22:41	**Cruise**	**SNG**	**ENG 2 COMP VANE OCCURRED AS PWR ADV TO CLB PWR THRU 17000 FT.**
13:41:01	*Cruise*	*ECR*	*ENGINE CRUISE REPORT*
13:42:21	*Cruise*	*CPR*	*CRUISE PERFORMANCE REPORT*
13:48:02	Cruise	FLR	ILS-2 NO DATA FROM CONTROL SOURCE // ILS 2
13:48:08	Cruise	FLR	AFS:FMGC2 // AFS,RMP1,2OR3
13:48:27	Cruise	FLR	AFS:ILS2 // AFS
13:55:50	Cruise	FLR	DME-1 TRANSCEIVER // DME 1
13:59:01	*Cruise*	*ECR*	*ENGINE CRUISE REPORT*
14:01:57	Cruise	FLR	DMC2: NO FQI2 DATA // EIS 2,EIS 1,EIS 3
14:37:23	Cruise	FLR	AFS:ADIRU1/2/3 DISAGREE // AFS
14:47	Cruise	Snag	SEAT COVER AT 3 F NEEDS TO BE CHANGED - COVER CHANGED
15:00:34	Cruise	WRN	NAV ALTI DISCREPANCY
15:08:54	Cruise	FLR	CHECK VHF-1 ANTENNA CIRCUIT // VHF 1
15:12:31	At Gate	MPF	MAINTENANCE POST FLIGHT REPORT
22:52		**Snag**	**REPLACED MASTER BALLSCREW ACTUATOR FLEX DRIVE &SPACER, ENG RUN CARRIED OUT,ACFT CHECKS SERV**

Although a simple example, it demonstrates the utility of such systems. Approximately 2 hours before arrival, a significant snag is picked out from all the message traffic and ground personnel are alerted. Parts can be ordered and a repair can be adeptly carried out with minimal disruption.

It should be noted that not all messages produced during the flight can be associated with this particular snag. Some might be generated by the aircraft spuriously or might indicate a new or ongoing snag. Others, such as the reports generated at 13:41:01, 13:42:21, and 13:59:01 (italics) represent routine tables of data recorded at certain flight phase and stability conditions which go into the *performance database*. Future versions of IDS will use these data for prognostics.

Current Situation

Four versions of IDS have been "released", each exhibiting substantial changes. Since the objective of this paper is to describe the current situation, the first three are only briefly outlined with further details found in [Wylie et al. 1997].

Version 0.1. This consisted of a set of interfaces (4 basic screens). It was partially able to access data provided on tape by Air Canada. Its main objectives were to confirm the pre-

liminary user interface design and to pull concrete ideas from the "clouds" to help set the direction.

Version 1.0. This was an off-line prototype which was able to "playback" and reason with one year's worth of data stored in a database. This had the benefit of permitting high volumes of data (at about 40 times real-time) to be pushed through IDS for testing purposes.

This version dealt with a subset of aircraft systems (flight controls and engines) due to unavailability of electronic manuals. Evaluators at Air Canada could now get a better feel of the system since it could replay conditions over a selected time period.

Version 2.11. This version has been in place at a number of sites in the airline since 1996. It interprets live data, is extended to the full aircraft (in this case the Airbus A320), has full access to the on-line manuals (troubleshooting and minimum equipment list), and has a modest case-base of historical problem resolutions.

After extended evaluation of this prototype at selected test sites, Air Canada made a decision to further develop and deploy the system. In addition to supporting this, NRC wished to continue towards the larger IDS vision. This meant dealing with the realities and goals of the two partners. Air Canada had to meet the needs of its users and deal with corporate realities such as communications infrastructure, budgets, manpower, and program standards.

Much of the effort in producing the v2.11 prototype had little to do with research. However, it was essential to place a stable and useful system within Air Canada and provide project credibility. During the next stage, NRC has committed to continued support of IDS v2.11 so that test sites can remain operative - including data redistribution via a server.

Towards IDS version 3.0

To set the stage for continuing research it was clear that the monolithic system that was IDS v2.11 required a redesign. It was the result of the project's history, various limitations, and changing requirements. As detailed in Figure 4, each user of the system was required to run the full IDS application, with each user application receiving the real-time aircraft messages, performing the identical reasoning and creating identical database records.

This obviously has a number of shortcomings:

- it does not scale well to large numbers of users,
- one cannot easily add modified or new interfaces as differing user needs are determined, and
- it does not provide a flexible research infrastructure. New functionality cannot be introduced without having a major impact on the existing application.

The natural solution was to distribute the various functions and services into a number of co-operating components. In order to build such a distributed system, commercial middleware software was used. Various solutions were examined for building such n-tier systems. These included Remote Procedure Call (RPC) based systems, Message Oriented Middleware (MOM) and Object Request Brokers (ORB). Ultimately a message oriented middleware solution from

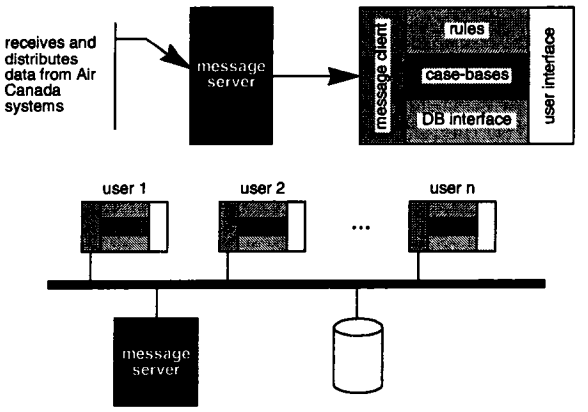

Figure 4. IDS version 2.11

Active Software, called *ActiveWorks* (see [Bracho and Green] or [PC Week 1998]) was chosen. It provides support for asynchronous delivery of typed messages (events) using either a publish/subscribe or a request/reply mechanism. It supports encryption of messages and various levels of message persistence to assist with guaranteed delivery. Messages are delivered in order and only once. There are also a large number of *adapters* available to facilitate integration with various languages (C, C++, Java, ActiveX, etc.) and commercial applications such as SAP, Varity, and PeopleSoft. The central component in an ActiveWorks distributed system is a *broker*, through which all events pass. There may be one or more brokers used in any application. The brokers and application components may be distributed on one, or many machines.

Figure 5 shows a simplified view of the IDS v3.0 components connected to a single broker (in practice several brokers on multiple machines are used). Briefly the function of

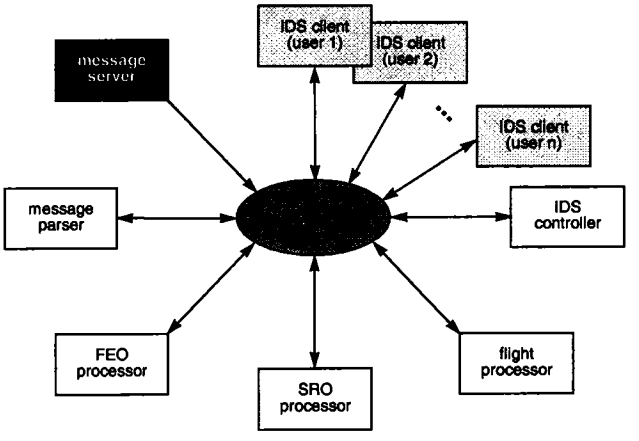

Figure 5. IDS version 3.0

each of the components of IDS v3.0 is as follows:

- **Message Server** - *receives* data/messages from Air Canada systems in real-time; *stores* these messages in text files for archive purposes; *maintains list* of *connected* IDS systems; *delivers* data/messages to connected IDS processes,
- **Message parser** - *receives* data/messages from Message server; *parses* messages forming ActiveWorks events;

publishes these parsed events,

- **FEO[1] processor** - *subscribes* to certain message published by Message Parser; *publishes* status of aircraft; *responds* to requests for status of aircraft; *responds* to requests for FEO information; *publishes* FEO information at end of flight leg,
- **SRO[2] processor** - *subscribes* to published FEOs; *responds* to requests for case-base searches,
- **Flight processor** - *subscribes* to messages published by Message Parser; maintains status of Air Canada flights; checks messages for accuracy with respect to aircraft and flight information,
- **IDS controller** - *controls* the startup and shutdown of the various IDS processes; *monitors* the state of the IDS processes (alive/dead, statistics, etc.), and
- **IDS clients** (users) - the function of these clients will vary depending on the view required of IDS information; may request aircraft status, subscribe to aircraft status events, request FEO details, request case-base searches, etc.

This architecture allows us to easily add new or experimental components into the application with little or no disturbance of existing components (as long as the new component is subscribing to published messages or requesting an existing service from one of the components). For example, it is a simple matter to add a component to monitor the distribution of messages and collect statistics or to add an experimental client component with a new user interface. In order to satisfy the requirement to maintain support for the IDS version 2.11 prototype, we modified the Message Server of version 2.11 to receive its raw messages from the IDS 3.0 Message Server rather than directly from the Air Canada systems. Thus, it just becomes another application that is connected to the IDS 3.0 server. This is shown in Figure 6.

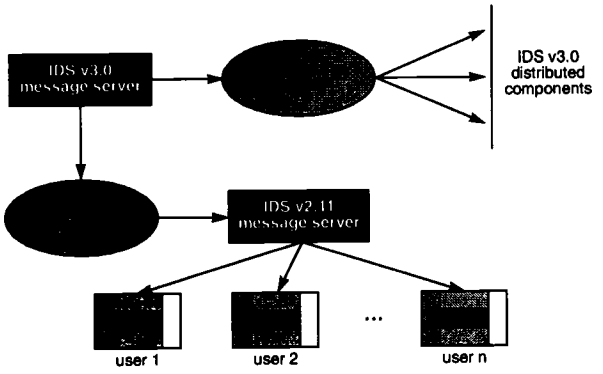

Figure 6. IDS 2.11 with IDS 3.0

Conclusion

There are two main issues worth discussing in the context of putting the theory into practice. The first is to highlight some of the obstacles facing adoption of this type of technology within a complex maintenance organization. The second is to discuss an alternate model of service delivery which may overcome some of these issues.

Technology Transfer Challenges

The challenges associated with technology transfer are considerable. One could well argue that they overshadow the technical issues. This section discusses practical issues which must be considered when taking such systems through the implementation phase.

It turns out that there is never an optimal time to initiate or deploy such systems. Even when projects such as these are economically justified, one must overcome a number of hurdles. The first is finding a strong internal champion and, to ensure that this role is properly transferred to their successors. These key individuals must embrace the vision and drive it through all the internal obstacles in order to build awareness, secure funding, and assemble a project team.

Critical Success Factors. When one considers successful deployment of systems such as IDS there are many factors to consider (see Figure 7). Some of the more critical questions

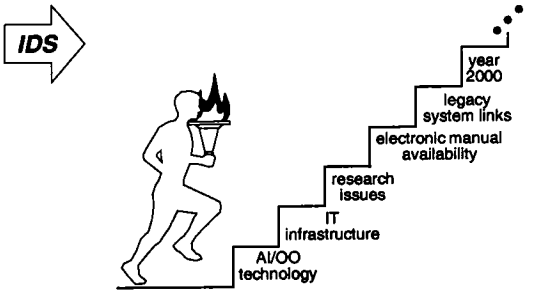

Figure 7. Deployment considerations

are:

- Can the *IT infrastructure* (LAN/WAN TCP/IP) perform adequately for "real time" data delivery and response at user locations? Can it be used to effectively deliver multi-tier distributed applications?
- Will an *electronic manual* system be available on-line at common workstations for IDS to index and reference?
- How will *partially resolved research* issues impact? For example:
  - an effective technique to automatically generate cases (without human intervention) to offset limited access to appropriate technicians, or
  - unsupervised learning techniques for predicting failures.
- Are the *required links to legacy systems* (e.g. flight movement) available and feasible?
- Is there an effective strategy to *minimize changes to business practices*? (i.e. evolution not revolution)?
- Has a process for *data "cloaking"* been determined?[3]
- Is IDS sufficiently *reliable*, *scalable*, and *maintainable*?
- Does the core software development team have *requisite skills* to deal with technologies new to the organization?[4]

1. Fault Event Object: the symptom cluster that IDS forms from the messages that are sent by the aircraft.
2. Snag Rectification Object: an FEO with the recommended associated repair appended to it - a potential basis for a case.
3. There is a requirement to temporarily "cloak" data on demand should one of the airplanes monitored be involved in an incident or accident. This requires compliance with Ministry Of Transport (MOT) regulations and Air Canada Quality Department
4. Note: in addition to the AI technologies, IDS also assumes the role of introducing object oriented techniques.

- Can the project obtain *active user participation*? The nature of the roles users play in the day to day operation is largely event driven. Users must be freed up to contribute to the project for uninterrupted periods - a very difficult exercise. Typical requirements are: requirements definition, user documentation, training, and front line support.

Alternate Service Model

When this project was initially conceived, the focus was on equipment operators. The general idea was for an organization to purchase software from a vendor and acquire upgrades and new functionality as they became available. In addition, the vendor or another company could provide integration and customization.

This is still a valid option that depends upon the goals and ambitions of the equipment operator. Equipment manufacturers, feeling the pressures of slim (at times nonexistent) profit margins on sales, have usually relied on parts and labour billing to recapture margins. Some companies are now pursuing another service model which assumes the burden of up-time guarantees - a form of maintenance outsourcing. Where one draws the line is difficult to predict. At one extreme, one could imagine an airline that doesn't have its own technicians or overhaul staff.

Our opinion is that there will be a spectrum of opportunities for service provision with airlines selecting a mix and match of service options which make sense for them.

Thus we believe that there is no right answer for how such systems should be deployed and delivered but it is worth considering another model being pursued. Figure 8 shows three roles:

- Air Canada continues as *lead user* - helps set directions, provides test sites, reaps benefits, and purchases service.
- NRC provides *R&D* to mitigate technical risk and help produce commercial software based on the IDS platform.
- The *service provision* role would be covered by a software developer and system integrator (which could be the same company):
  - The developer takes R&D prototypes and packages them into a product suite (could operate remotely from the operator) and develops the market.
  - The system integrator provides 7/24 support, maintains data integrity from a variety of airlines, retrieves data from appropriate databases, and distributes information back to users.

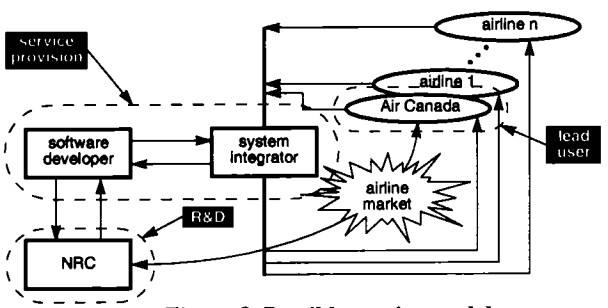

Figure 8. Possible service model

An important concept is a shared service subscriber casebase which will greatly improve diagnostic coverage. This avenue is under consideration by Star Alliance[1] members to allow seamless maintenance support between organizations.

Ongoing and Future Work

The project is once again evolving towards a longer term focus. The prototype, along with its infrastructure and data resources provides an excellent platform to test new ideas. Many important issues that require further work or remain unsolved are:

- automate case creation and maintenance. Work is ongoing to explore more refined ways in which to extract meanings from free form text;
- improve upon automated symptom clustering strategies and indexing of the case-bases;
- determine if the diagnostic coverage of the case-based reasoning paradigm is adequate when applied to the collective "experience" of Air Canada *or* if another operator's experience should be shared *or* if it is worthwhile to explore deeper reasoning techniques;
- experiment with model-based trend analysis;
- provide robust repair management advice by extending IDS to reason about other data sources and knowledge such as: downline station capability, component repair history/reliability, aircraft repair/maintenance history, deferred problems, parts location, schedules, and flight movement (including weather);
- extend to other Air Canada aircraft, such as Airbus 319/330/340, Canadair CL65, and Boeing 767;
- develop and test delivery modes for a variety of end users (e.g. Personal Communications Services (PCS), palm computing, wearable, pen computing solutions); and
- continue work on the **A**erospace **D**ata **M**iner (ADAM) prototype, a domain specific system that provides data mining and monitoring capabilities. Recent results have discovered very useful patterns from large amounts of data (text and parametric) to predict component failures and detect abnormalities. Field testing is in progress.

References

1. Bracho, R.; Green, J.; *Understanding the Integration Iceberg*, http://www.activesw.com/home.htm.
2. Canaan Group Limited, *Aviation Maintenance in the Information Age: Cutting Line Maintenance Costs with Information Technology*, Spring 1998, An Aviation Industry Commentary.
3. McConnell, S.; *Rapid Development: Taming Wild Software Schedules*, 1996, Microsoft Press.
4. PC Week, *ActiveWorks 3.0 hides the plumbing of app integration, letting administrators do their jobs*. New York, 1998, v.15, n.51, p.37, 3p. ISSN 0740-1604.
5. Statistics Canada, 1996. *Private and Public Investment in Canada*, Investment and Capital Stock Division, Revised Intentions, catalogue 61-206 Annual.
6. Wylie, R.; Orchard, B.; Halasz, M.; Dubé, F.; *IDS: Improving Aircraft Fleet Maintenance,* Innovative Applications of Artificial Intelligence, IAAI-97, July 27-31, 1997, Providence, Rhode Island.

1. Consists of: Air Canada, Air New Zealand, All Nippon Airways, Ansett Australia, Lufthansa German Airlines, SAS - Scandinavian Airlines System, Thai Airways International, United Airlines, and VARIG Brazilian Airlines.