Just IN.

The development of early software based telecommunications services, up to and including the 'Intelligent Network'.

Cliff.Redmond@cs.tcd.ie, Vincent.Wade@cs.tcd.ie

This technical report traces the development of software based telecommunications services from the 1960's to the 'Intelligent Network' (IN) initiative of the late 1980's and early 1990's. It presents an outline history of the IN initiative, together with its objectives and architecture. The report concludes by evaluating the role of IN as a software architecture for telecommunications services.

1 Early telecommunications service software

The introduction of computer technology into telecommunications started in the 1960's when analogue switching systems had their common control facilities replaced by a 'control program' (common control refers to the use of a pool of registers to facilitate number translation and hence routing). These 'stored program control' switches initially replaced facilities implemented using electromechanical relays. Later as fully electronic switching systems based on digital switching techniques became available [Ithell89], it became easier to extend and amend the 'control program' to provide services over and above basic telephony to telecommunications subscribers. The replacement of electromechanical hardware with programmable controllers thus facilitated easier introduction of new services, i.e. hardware modifications were no longer necessary.

The advent of centralised control put new requirements on the reliability of the control system (both hardware and software), with at least duplication of facilities to provide the redundancy necessary to counteract catastrophic failures causing loss of service [Littlechild79]. The importance of the control program and the implications of its failure are clear; software reliability is paramount if the high standards of reliability and availability the telecommunications community has set itself are to be maintained; for example, an allowance is made for a maximum of two hours outage every forty years! [Scherer88].

A variety of 'switch based services' soon followed to take advantage of facilities made easily accessible by software control. These early services mirrored previous hardware based switch services and were normally just functional extensions to enable service providers to extract more value from existing services like POTS (plain old telephone service); Call Waiting and Call Answering are examples of such services. These facilities were, and still are, the basis of 'switch based services'. In this scenario manufacturers were in control of the development and delivery of new services [YoungJ88], and could virtually hold operators to ransom for new service development costs because operators were locked into using what were indispensable propriety systems [Shapiro99].

2 The 'Intelligent Network'

The 'Intelligent Network' (IN) was first described in 1985 [Hass88]; it was the first significant attempt by the telecommunications industry to specify a service platform and architecture. While its services can be seen as quite basic in software terms, IN was the first standards based approach to facilitate software based service deployment on public telecommunications networks. The IN architecture still underlies many current telecommunications service implementations, e.g. GSM location registers [TINAB99].

2.1 Background and history

As competition was introduced into telecommunications, first in the United States and later in other markets, service providers found it necessary to speedily deploy new services to differentiate themselves from their competitors and to respond to customer requirements. Service providers found themselves increasingly at the mercy of equipment manufacturers ('switch vendors') who remained in total control of the development and deployment of new services [YoungJ88]. Often there was a clash of interests: the equipment manufacturer could be offering subscriber products which were substitutes for the very services that service providers wanted them to implement! An example of this is the 'Centrex' service, where local service providers offered a bundle of services providing a 'virtual' PBX (private branch exchange) and equipment manufacturers offered actual PBXs installed on a subscriber's premises.

In response to what could charitably be regarded as tardiness on the part of their equipment manufacturers, several regional Bell operating companies (companies providing local telecommunications services to regions of the United States, i.e. service providers) joined with Bellcore, their central research and standard setting body, to produce a standards based service architecture; the 'Intelligent Network'. Standards were seen by operators as a means to extract themselves from their 'lock-in' to specific switching platforms by promoting competition between manufacturers and facilitating manufacturer independent service creation and deployment.

The term 'Intelligent Network' is a blanket term for a number of telecommunications industry and standards body initiatives that started in the mid 1980's. The first intelligent network proposals owed their existence to an *ad hoc* industry consortium [Hass88]; this produced an architecture subsequently known as IN/1 in 1985. IN/1's intent was to facilitate rapid service introduction by separating service logic from switches, however the major software elements were service specific and needed to be developed anew for each new service [Gansert88]. IN/2 followed soon after in late 1986.

In its time, the 'second generation' intelligent network was billed as a near panacea; it was specifically designed to enable the rapid development and deployment of new services [Gansert88], [Berman92]. It augmented the IN/1 architecture by expanding the set of switch and service capabilities, defining 'Functional Components' (FCs) to facilitate rapid service creation and adding better user interface capabilities through an 'Intelligent Peripheral' connected to the switch. It was soon realised that IN/2 was "an overly ambitious proposal which would entail unacceptably high risks and could not be implemented in a sufficiently short time" [Berman92]. The risks were mainly financial, not technical.

Based on this realisation, Bellcore released the IN/1+ architecture; this incorporated the 'Functional Components' and 'Intelligent Peripheral' of IN/2, but used just a subset of the functions specified - those needed for voiceband services. IN/1+ was to:

- Be attainable by 1991.
- Offer a profitable set of service opportunities.
- Offer an evolutionary step towards IN/2 [Gansert88].

As a short term solution, IN/1+ soon revealed its limitations; concern grew about the load on the signalling network that was also used to support the new services' switch independence [Pierce88].

In the late 1980's the IN/1+ initiative was shelved, and an industry forum called the Multi Vendor Interaction (MVI) was convened [Berman92]. This was launched in early 1989 and produced standards in 1990; the initial standards called Advanced Intelligent Network (AIN)

Release 1, unified the many IN/1 architectures (relabelled 'Release 0' architectures) which by then existed. 'AIN Release 1' was an aspirational standard; it was seen as a model towards which the IN would evolve - its architecture was highly correlated with IN/2. 'Release 0.1' and 'Release 0.2' were introduced as realisable intermediate architectures towards 'Release 1'; 0.1 introduced a common formal call model and 0.2 ISDN (integrated services digital network) announcement capabilities and default routing amongst other features. AIN Release 2 was announced in 1995.

Concurrent with these standards set in the United States' telecommunications market, the CCITT (*Comité Consultatif Internationale Télégraphique et Téléphonique*, now the ITU-T International Telecommunications Union – Telecommunications Services Sector) in conjunction with ETSI (the European Telecommunication Standards Institute), attempted to standardise the IN for other service providers [Garrahan93]. It defined Capability Sets (CS), with CS-1 roughly equivalent to AIN Release 1, and constructed a coherent IN conceptual model [Duran92]. Bellcore has since migrated its standards to use the terms defined in the ITU-T standards.

2.2 Objectives

The IN initiatives attempted to meet the following objectives:

- **The creation of a market for telecommunications equipment**

By specifying a minimum functionality that they expected telecommunications equipment to have, telecommunications service providers hoped that price based competition between equipment vendors would help drive their costs down.

- **Rapid service introduction**

The key objective of IN/1 was "the ability to rapidly and flexibly add new services without requiring upgrades to the embedded switching system software" [Homayoon88]. IN/1 facilitated rapid service introduction by allowing centralised service deployment into a telecommunications network [Gilmour88], covering a "wide geographic area" (namely a U.S. regional operator or other administrative domain); it also allowed services to use the facilities of a number of switches.

It was thought that the 'Service Control Point' (SCP), the primary architectural element enabling this flexibility, would provide a suitable basis for the introduction and evolution of new services, but that ultimately, when proven (both technically and economically) the services would be incorporated into the switch [Head88] due, amongst other things, to the performance overhead added by using a separate service control point.

- **Service independent capabilities**

IN/2 sought to "reduce the interval between new service introduction", i.e. to reduce the time needed to create new services. Its key means to realise this objective was the specification of service independent capabilities that could be re-used in the definition of new services. A subset of these capabilities was adopted for IN/1+ [Bauer88].

- **Easier service creation**

Ultimately, it was recognised that it was better to support service creation at service provider rather than manufacturer level [Pinkham88], [YoungJ88], [Bauer88], [Berman92] and that "the success of IN [depended] on the ability of operator company personnel to create new services" [Pierce88].

2.3 Architecture

The key elements of the IN architecture are outlined in Figure 1. This figure uses the symbols commonly used to describe the IN architecture as used in [Gilmour88], [Weisser88] and [Berman92], and combines the elements described therein.

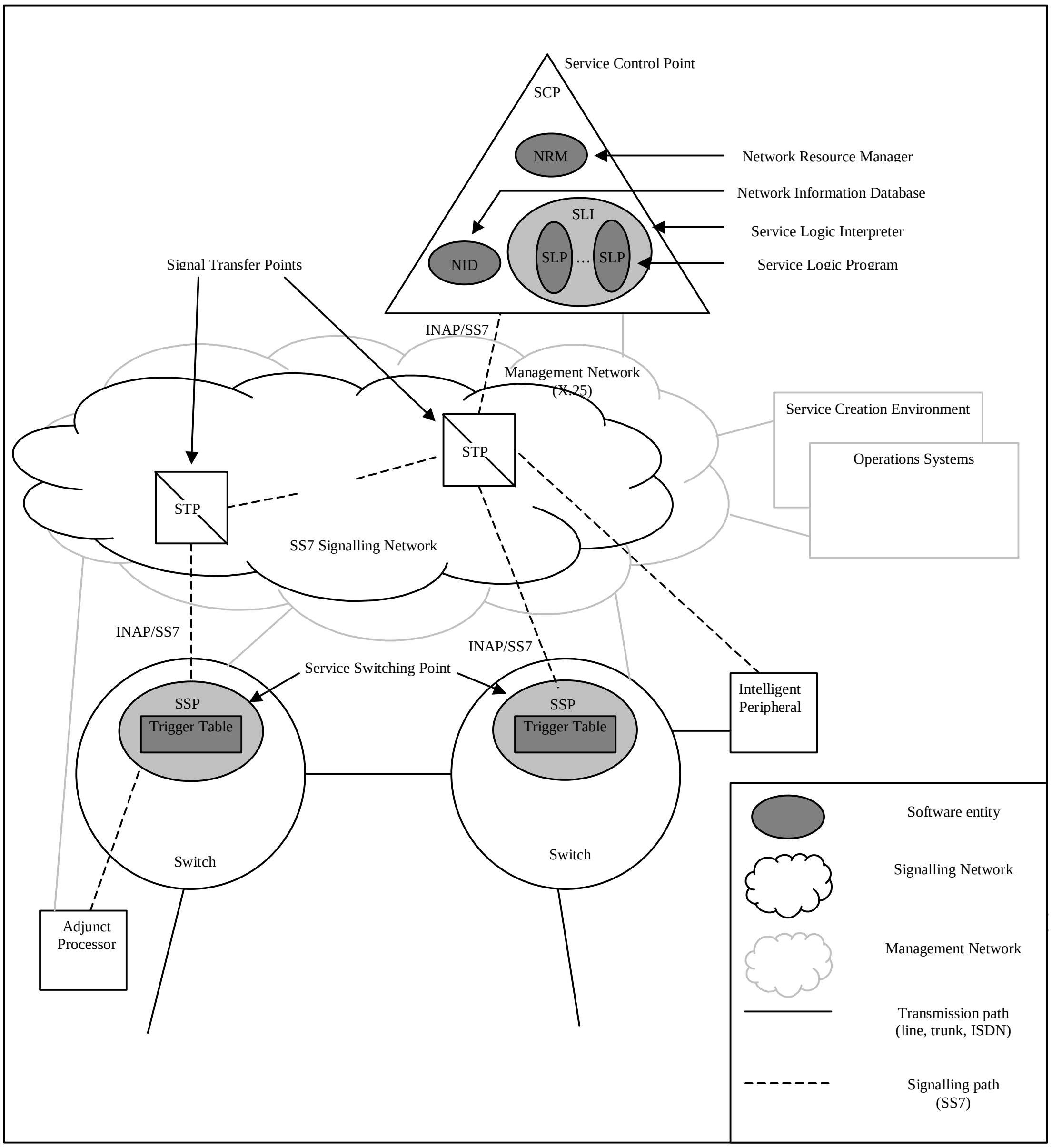

Figure 1, The IN architecture

A brief explanation which outlines the function of each element in Figure 1 follows. It has been noted that the relative simplicity of the IN is obfuscated by the abundance of acronyms which appear to serve to confuse the "uninitiated"! [Mercury93]. In deference to this, the explanations offer approximately equivalent, and more common, terms for many of the elements described.

The invocation of an IN service starts by the detection of a 'trigger', or event, at predefined 'trigger detection points' within the call (the specification of these detection points thus required the definition of a generic IN call model). When a trigger is detected, normal call processing is suspended, and information in the trigger table is used to formulate a query to the 'Service Control Point' (SCP). The query is expressed in a special purpose protocol (the

IN Application Protocol (INAP)) then sent to the SCP over the signalling network, through 'Signal Transfer Points' (STPs), or packet switches.

The 'Service Control Point' processes this query and can either return a set of instructions to the 'Service Switching Point' in the switch, or execute them in its own 'Service Logic Interpreter' (SLI), with performance and signalling network traffic implications influencing the choice of execution environment [Gilmour88]. The set of instructions is called the 'Service Logic Program' (SLP). (The 'Service Logic Interpreter' was defined in IN/2 - prior to his IN services did not share control facilities; each service had its own 'Service Control Point' [Gansert88]).

The instructions consist of a set of 'Functional Components' (FCs) to be executed for the call, these are defined as "elemental network call processing actions" [Gilmour88]. (Example instructions include those to control call 'legs' (e.g. *create*, *join*, *free*) and those to give and receive information from call participants (e.g. *send* and *receive*) [Bauer88]). The 'Functional Components' were the key to service independence; as primitives they were designed to be reusable at quite a high level and well defined enough to be externally invokable on a 'Service Switching Point' (SSP) [Bauer88].

The other elements in the architecture include the 'Adjunct Processor': this is similar to the 'Service Control Point' except that it is not accessible through the signalling network, and is therefore not shareable: it has a direct communication link to the switch, and the 'Intelligent Peripheral', which offered an "enhanced" user interface to IN services through voice synthesis, announcements, speech recognition and digit collection [Berman92]. Logical elements included the 'Network Information Database' (NID) which kept information about access lines and trunks, and the 'Network Resource Manager' (NRM) which provided a location function enabling the correct switch to be invoked to continue call processing [Gilmour88].

The 'Service Creation Environment' provides an environment to compose IN services from 'Functional Components', it is usually present in an IN administrative domain, but not subject to IN standardisation - it is normally supplied by the SCP vendor, i.e. it is proprietary [MaA94a], [Kockelmans95].

'Operations Systems' offer support and management services variously called Operations, Administration and Maintenance (OA&M) [Bauer88], Operations, Administration, Maintenance and Provisioning (OAM&P) [Glitho95] or simply Operations Support Systems (OSS) or Operations Systems (OS).

2.4 Evaluation

This section outlines the achievements and deficiencies of the 'Intelligent Network' architecture as defined and deployed in telecommunications networks throughout the world.

IN is technology specific. Most IN telecommunications services can be regarded as supplements to basic telephony service [Hellemans96]. In common with other telephony services, IN services must suffer the limited nature of their user interface [MaA94a]. Service processing is dependent on the detection, by the 'Service Switching Point' within the switch, of 'triggers' at 'trigger detection points' in the context of a 'call' prior to connection set-up, i.e. services are invoked for the end user by the transport provider [LaPorta97]. Within the switch, the call is modelled by either of two finite state machines: the Originating and Terminating Basic Call Models (BCM) [Berman92], which are, by their nature, strongly telephony oriented - states include those for routing, set-up and release. Each state is called a 'point in call' (PIC) and has an associated detection point where call processing can be handed over to a 'Service Control Point' (SCP).

The ‘Service Control Point’ has a standardised, generic, ‘Connection View’ of the call processing resources an IN switch offers [Berman92]. This standard model, while enabling some switch vendor independence, offers little in the way of transport technology independence to IN services; this was evidenced by the difficulty faced in modelling multiparty calls for IN Capability Set 2 (CS-2) - to enable such IN services as call waiting, call transfer and conference calls - because they necessitated changes to the connectivity of an existing call [O’RR98].

The physical separation of the ‘Service Switching Point’ (SSP) and the ‘Service Control Point’ (SCP), attempted to provide the reliability required for telecommunications services by provisioning expensive centralised facilities. IN did not attempt to distribute the service itself [Barr93] either by endeavouring to build a reliable system by redundantly deploying relatively cheap and unreliable facilities, or by any other means. This was reasonable given the state of the art in distributed systems at the outset of IN standardisation [Head88], but ultimately it leaves the IN looking like a legacy centralised system, by its nature more prone to catastrophic failure than a counterpart with distributed intelligence.

While failing to achieve a complete logical separation between a service and its underlying communications technology realisation, IN nevertheless established a physical separation between a service and its delivery [Hellemans96] that provided a useful basis for further service modelling work.

IN succeeded in creating a market for telecommunications equipment through standards, but detractors consider the inflexibility caused by over-specification in these standards acts to stifle innovation in offering new services [Isenberg98].

IN partially succeeded in its attempt to enable service providers to define their own services; switch vendors now sell ‘Service Creation Environments’ that take advantage of the ‘service independent’ facilities defined by IN. While it is unknown whether the overall costs of services developed using these environments is significantly less than the cost of purchasing services previously developed by switch vendors, service providers have been empowered to create and deploy their own services, albeit within the confines of proprietary environments [MaA94a], the currently deployed version of the IN Application Protocol (INAP) [TINAB99], and other technology constraints.

One of the reasons for the limited success of ‘Service Creation Environments’ was the advent of ‘feature interactions’ between IN services. In this context a ‘feature’ was either a service constituent [Zave93] or a simple service itself, used as a synonym because the term ‘service’ had become overloaded [Cameron93a]; similarly, the term ‘service interaction’ referred to the interaction of IN and switch based services (i.e. pre-IN services wholly implemented within the switch).

New services introduced into the public telecommunications network showed unwanted and adverse interactions [Griffeth93], [Zave93], [MaA94a], i.e. where the use of one service was altered by the use of another; for example call forwarding and call waiting [Cameron93b]. The problem was variously seen as one of incomplete system specification [Zave93], [Cameron93a] [MaA94a], of software re-use and maintenance [Griffeth93], [Cameron93a], of distributed systems (timing and race-conditions) and of artificial intelligence (dynamic resolution of conflicting end-user needs) [Cameron93a]. IN still exists, but the need to solve the problem of adverse interactions between service features has led to the adoption of contemporary computer science techniques, in a combined application of the approaches mentioned above, towards the definition of a wholly software based service architecture [TINAB99].

‘Service Creation Environments’ are also limited by the effects that the underlying service transport mechanism has on the flexibility of service definition. IN services are designed to use a dedicated underlying transport protocol (i.e. SS7), with the asynchronous IN Application Protocol (INAP) interactions facilitating the soft real time requirements of the signalling network. Unfortunately this requirement, while providing reliability, restricts the flexibility of service definition [Hellemans96].

IN neglected operations aspects of service provision, failing to specify standardised functional components that could be used to manage IN services. While the importance of management aspects was acknowledged [Bauer88], the IN initiative concentrated standardisation effort on service switching and control [Kockelmans95]. The creation of a software dichotomy ultimately proved harmful; later telecommunications service architectures recognised the need to support the concurrent development of services and their management facilities [TOCP95].

2.5 References

[Barr93] William J. Barr, Trevor Boyd, Yuji Inoue. The TINA Initiative. *IEEE Communications Magazine*, 31(3): 70-76, March 1993.

[Bauer88] Helen A. Bauer, John J. Kultzer, Edward G. Sable. Designing Service-Independent Capabilities for Intelligent Networks. *IEEE Communications Magazine*, 26(12): 31-41, December 1988.

[Berman92] Roger K. Berman and John H. Brewster. Perspectives on the AIN Architecture. *IEEE Communications Magazine*, 30(2): 27-33, February 1992.

[Cameron93a] E. Jane Cameron, Nancy Griffeth, Yow-Jian Lin, Margaret E. Nilson, William K. Schnure and Hugo Velthuijsen. A Feature-Interaction Benchmark for IN and Beyond.

[Cameron93b] E. Jane Cameron and Hugo Velthuijsen. Feature Interactions in Telecommunications Systems. *IEEE Communications Magazine*, 31(8):18-23, August 1993.

[Duran92] José M. Duran, John Visser. International Standards for Intelligent Networks . *IEEE Communications Magazine*, 30(2): 34-42, February 1992.

[Gansert88] Joseph A. Gansert. Intelligent Network Evolution Issues. *IEEE Communications Magazine*, 26(12): 59-63, December 1988.

[Garrahan93] J.J. Garrahan, P. A. Russo, K.Kitami and R. Kung. Intelligent Network Overview. IEEE Communications Magazine, 31(3):30- 38, March 1993.

[Gilmour88] Julia Gilmour and Richard D. Gove. Intelligent Network/2: The Architecture – The Technical Challenges – The Opportunities. *IEEE Communications Magazine*, 26(12): 8-11, December 1988.

[Glitho95] Roch H. Glitho, Stephen Hayes. Telecommunications Management Network: Vision vs. Reality. *IEEE Communications Magazine,* 33(3): 47-52, March 1995.

[Griffeth93] Griffeth, Extending telecommunications systems: The feature-interaction problem, *IEEE Computer* August 1993

[Hass88] Ronald J. Hass, Stewart D. Personick. Working with the Industry to Shape Intelligent Network Requirements. *IEEE Communications Magazine*, 26(12): 12-15, December 1988.

[Head88] Charles S. Head. Intelligent Network: A Distributed System. *IEEE Communications Magazine*, 26(12): 16-20, December 1988.

[Hellemans96] Patrick Hellemans, A Comparison Between TINA and Legacy Signalling, In *Proceedings TINA'96 Conference* (ISBN 3-8007-2201-1), VDE-VERLAG, Berlin & Offenbach, 1996.

[Homayoon88] Seth Homayoon, Harmi Singh. Methods of Addressing the Interactions of Intelligent Network Services with Embedded Switch Services. *IEEE Communications Magazine*, 26(12):42-46, December 1988.

[Isenberg98] David S. Isenberg. The Dawn of the Stupid Network. Mixed Media. ACM Networker. February/March 1998.

[Ithell89] Alan H. Ithell. Service Integration on Digital Networks. Cimtech (Hatfield Polytechnic), 1989.

[Kockelmans95] Maarten Kockelmans, Eric de Jong. Overview of IN and TMN Harmonization. *IEEE Communications Magazine*, 33(3):62-66, March 1995.

[LaPorta97] Thomas F. La Porta and Kuo-Wei Herman Chen. A Direct Signalling System for Flexible Access and Deployment of Telecommunications Services. *IEEE/ACM Transactions on Networking*, 5(4): 489-501, August 1997.

[Littlechild79] Stephen C. Littlechild. Elements of telecommunications economics. Institution of Electrical Engineers, London., 1979.

[MaA94a] Mícheál Mac an Airchinnigh and Hans-Jürgen Kugler. Service Engineering Versus Software Engineering – A Foundational Study. In *Proceedings of the International Conference on Intelligence in Services and Networks (IS&N'94),* Springer-Verlag 1994.

[Mercury93] Mercury Communications. The Magic of IN. Mercury Communications Ltd.
Available at http://www.gare.co.uk/technology_watch/in.htm
February 1993.

[O'RR98] Maureen O'Reilly Roche. Call Party Handling Using the Connection View State Approach: A Foundation for Intelligent Control of Multiparty Calls. IEEE Communications Magazine, 36(6):60-66, June 1998.

[Pierce88] Michael Pierce, Fred Fromm, Fred Fink. Impact of the Intelligent Network on the Capacity of Network Elements. *IEEE Communications Magazine*, 26(12): 25-30, December 1988.

[Pinkham88] Gary Pinkham, Intelligent Network Feature Transaction Interface. *IEEE Communications Magazine*, 26(12): 47-52, December 1988.

[Scherer88] Joan, R. Scherer, Thomas A. Murray. Management of the Intelligent Network. *IEEE Communications Magazine*, 26(12): 21-24, December 1988.

[Shapiro99] Carl Shapiro and Hal R. Varian. Information rules: A Strategic Guide to the Network Economy. Harvard Business School Press, Boston Mass.,1999

[TINAB99] Hendrik Berndt et al. The TINA Book. Prentice Hall Europe, 1999.

[TOCP95] Martin Chapman (ed.). Overall Principles and Concepts of TINA. TINA Consortium Feb. 1995.

[YoungJ88] John W. Young. Service Creation: A Telephone Company Perspective. *IEEE Communications Magazine*, 26(12): 64-70, December 1988.

[Zave93] Pamela Zave. Feature Interactions and Formal Specifications in Telecommunications. *IEEE Computer*, 26(8):20-29, August 1993.