

Capítulo V: Trabajo cognitivo y control sobre el proceso de trabajo: la producción de software

J. Guadalupe Rodríguez Gutiérrez¹

Enrique de la Garza Toledo

I. Producción material, inmaterial y simbólica.

Todos los trabajos tienen aspectos materiales e inmateriales, objetivos y subjetivos, al grado de haber producción eminentemente simbólica tanto en su aspecto objetivo como subjetivo. Una complicación es la intervención del cliente, que puede darse en la producción material o inmaterial. En el primer caso, cuando el producto adquiere un carácter objetivado, incluyendo la posibilidad de la producción eminentemente de símbolos objetivados; en el segundo, que se produce con la intervención del cliente en la que el producto se consume en el momento de la producción.. Todas estas posibilidades de la producción, material-inmaterial, simbólica objetivada o subjetivada abren la necesidad de repensar conceptos clásicos de la Sociología del Trabajo para analizar procesos de trabajo.

1.) Primero, en cuanto al control sobre el proceso de trabajo. Este concepto se popularizó en la sociología del trabajo en los años cincuenta, sesenta y setenta y la imagen detrás era la del trabajo taylorizado, muy controlado, frente al trabajo del asalariado de oficio o bien del artesano autoempleado. En el concepto de control clásico influyó mucho la ciencia política norteamericana de los sesenta que lo entendió a la manera weberiana como capacidad de imponer la voluntad a otro. En esta medida, en el proceso productivo se podrían controlar los insumos, la maquinaria y el equipo, la distribución del espacio físico, el tiempo de trabajo, las operaciones a realizar, la calificación, los conocimientos, las interacciones dentro del trabajo, la cultura y la subjetividad, las relaciones laborales (entrada y salida al trabajo, ascensos, salarios y prestaciones, capacitación, afiliación a sindicatos, a los sindicatos mismos). En la perspectiva marxista clásica, el problema del control con Braverman se remitía a un nivel más general, en la producción capitalista para explotar al trabajador el capital tenía que dominar al obrero dentro del proceso de trabajo. Es decir, esta concepción remite, antes que al control, al problema clásico de la ciencia política del poder y la dominación pero en el proceso productivo. Entendidos clásicamente como poder coerción y dominación como consenso y en un caso extremo Hegemonía. En general, el ámbito del poder puede implicar la imposición personalizada o bien

¹ joserodriguez@nogales.uson.mx ; Profesor en la Universidad de Sonora, Unidad Nogales. Doctor en Estudios Sociales por la Universidad Autónoma Metropolitana, Iztapalapa. Enrique de la Garza Toledo, profesor-investigador de la UAMI: egt@xanum.uam.mx

abstracta a partir de reglas organizacionales o técnicas. La dominación se acerca a la legitimidad del mando y en el extremo al reconocimiento por parte de los dominados de esa capacidad intelectual y moral de ser dirigidos por la gerencia. Es decir, hunde sus raíces en el campo de la cultura y la subjetividad, en la construcción de los sentidos del trabajo y de las relaciones entre las clases en los procesos productivos que supone los niveles cognitivo, moral, estético, sentimental y a las formas de razonamiento cotidianas, sintetizadas en discursos y formas de conciencia no discursivas. La hegemonía, como en general el poder y la dominación pueden ser todas concebidas como construcciones sociales que implican estructuras de los procesos de trabajo y fuera de estos, interacciones inmediatas y mediatas, formas de dar sentidos en la producción y en otros espacios de interacciones y niveles de realidad..

Es decir, el poder y la dominación en el proceso de trabajo que puede sintetizarse en constelaciones tales como clientelismo, patrimonialismo, caudillismo, dominación burocrática, patriarcalismo, democracia, dictadura, etc., pueden operacionalizarse a través del concepto de Control Pero el control tiene que especificarse en que tipo de relaciones de producción se concreta –de explotación, de autoempleo, de subcontratación, etc. En este camino de lo abstracto del poder y dominación en el proceso de trabajo hacia el control con varias dimensiones, las tipologías del debate del proceso de trabajo pueden ser útiles –control técnico, administrativo, autocontrol- sin olvidar que todas las formas de control implican ciertas interacciones igualitarias o bien jerárquicas, que estas interacciones ponen en juego símbolos y sus intercambios o negociaciones, que implican reglas formales e informales, relaciones con determinadas estructuras.

Remitiéndonos a los grandes tipos del trabajo no clásico (De la Garza, et.al, 2010). En el primero que se realiza en espacios cerrados y con trabajo asalariado aunque con intervención de los clientes, el problema del control empieza por el que puede ejercer el empleador y la cooperación o resistencia que pueden hacer los trabajadores. En esta dimensión caben los aspectos mencionados para el control en el trabajo capitalista. Pero al que hay que añadir el que puede ejercer en mayor o menor medida el cliente. El control del cliente empieza como presión simbólica para la realización del trabajo en el espacio y tiempo esperado y con la calidad requerida por este. En este control pueden jugar las reglas organizacionales esgrimidas por el cliente u otras más amplias de la legislación, incluyendo los derechos humanos, hasta el recurso de la apelación a la ética combinada con las emociones. Esta presión puede traducirse en interacciones, peticiones, reclamos hasta la demanda legal, según el caso, sin excluir el uso de la fuerza física o simbólica por parte del cliente o del trabajador.

La situación se complica para los trabajadores en espacio abierto de interacciones con sujetos diversos, sea en locales fijos o no (trabajo no clásico de tipo II). Cuando se trata de trabajo asalariado

valen las consideraciones ya expresadas para esta dimensión obrero patronal, habría que agregar la relación con el cliente. Pero en este caso no se puede hablar de la participación de este en un “contrato” de trabajo al cual se pueda apelar, como el derechohabiente del seguro social que puede reclamar un servicio previamente pactado, sino que en el mejor de los casos valen reglas más generales del derecho civil, del penal, del reglamento de policía o del de salubridad. Pero lo que lo hace francamente complejo es la emergencia no necesariamente sistemática de actores del territorio que no implican la relación proveedor-trabajador-cliente, estos actores pueden ser transeúntes, policías, inspectores, otros trabajadores de la misma ocupación, líderes de organizaciones de este tipo de trabajadores o de otras actividades (De la garza, 2007). Aunque estas relaciones no son de trabajo impactan al trabajo, impactan el uso del territorio para trabajar, al tiempo, al tipo de producto, a las ganancias, y hasta a la existencia misma de la ocupación. Aunque son interacciones eventuales no necesariamente son extraordinarias y muchas veces es posible establecer regularidades en cuanto al tipo de actor que interacciona, el tipo de interacción, los contenidos prácticos y simbólicos, las cooperaciones, negociaciones y conflictos.

El tercer tipo de trabajo no clásico (tipo III), en espacios cerrados, privados, con interacciones precisas con patrones, proveedores y clientes, lo que añade complejidad en cuanto al control son las presiones que vienen de la familia, las interfases y a la vez contradicciones entre espacios de trabajo y los de alimentación, comida, aseo, cuidado de los niños, descanso o diversión. Otro tanto se puede decir de las interfases y contradicciones entre tiempos productivos y reproductivos y entre las mismas actividades. Los actores adicionales a considerar son los hijos, esposos, familiares que cohabitan en el mismo espacio de trabajo y que exigen atención, tiempo, afecto, otro tipo de trabajo el doméstico para sus necesidades vitales.

2.) La regulación laboral. Este tema ha estado asociado en el origen al surgimiento del trabajo asalariado sin regulaciones ni protecciones en los trabajadores de los siglos XVIII y XIX. Las luchas obreras fueron conformando un cuerpo de Leyes, contratos, etc. que regularían estas relaciones en cuanto al salario y prestaciones, la entrada y salida de los trabajadores del empleo, los tiempos de trabajo, las funciones a desempeñar, la calificación necesaria, la forma de ascender entre categorías, las de cómo resolver las disputas entre capital y trabajo, las sanciones a los trabajadores cuando incumplan las normas, la movilidad interna, la polivalencia, los escalafones, la participación en las decisiones anteriores de los trabajadores o de los sindicatos en los cambios tecnológicos o de organización. Lo anterior se extendió más allá del lugar de trabajo hacia las instituciones de seguridad social y a las de la justicia del trabajo, así como a los vínculos más amplios entre Estado, sindicatos y organizaciones patronales. Todo esto se consideró por mucho tiempo que solo era pertinente para el trabajo asalariado,

por la razón de que se podría demandar por el incumplimiento de las normas a los trabajadores o a las empresas, a diferencia del autoempleado que al no contar con un patrón se consideró ausente de una relación laboral. Sí bien la relación laboral en sentido restringido puede ser la que se establece entre el capital y el trabajo, que parte del puesto de trabajo y que se extiende hasta el Estado y las instituciones de justicia laboral y de seguridad social. En sentido ampliado la relación laboral no sería sino la o las relaciones que en el trabajo se establecen entre los diversos actores que participan interesada o circunstancialmente en este y que influyen en el desempeño laboral. Con esta definición ampliada de relación laboral como interacción social, con sus atributos de práctica e intercambio de significados en determinadas estructuras, los actores a considerar no tendrán que ser solamente los que contratan fuerza de trabajo y quienes son contratados, dependiendo del tipo de trabajo pueden ser actores muy diversos no necesariamente interesados en la producción de un bien o un servicio determinado (De la Garza, 2007^a).

En el primer tipo de trabajo no clásico, tipo I, (asalariados en espacios cerrados en interacción directa con los clientes) vale la pena detenerse en la parte correspondiente al cliente. El buen trato al cliente es parte muchas veces de lo pactado en la relación laboral como asalariado de este tipo de trabajador, el interaccionar con este actor con eficiencia y con cortesía puede ser parte de la regulación formal. Frente a un incumplimiento de esta norma el cliente puede acudir a la empresa o a otras instancias de regulación –por ejemplo de los servicios médicos- para demandar al trabajador, eventualmente puede también recurrir al derecho civil o al penal. Pero lo más interesante podría ser la regulación informal que apela a la ética del trabajador en la atención, la cortesía o a las buenas costumbres. Dependiendo del caso también pueden influir sobre el trabajador sentimientos de compasión (discapacitados) y la presión simbólica y hasta física de otros clientes que hicieran cusa común con los reclamos. La presión del cliente puede encontrar apoyo en reglas de la organización que emplea al trabajador- las horas de entrada se convierten en inicio de la atención para el cliente, igual que las horas de salida o los poros en la actividad del trabajador motivos de disputas. Es decir, las reglas burocráticas de la empresa pueden ser usadas por los usuarios y con esto adquirir un carácter tripartito en la práctica. Otro tanto se puede decir de reglas sancionadas directamente por el Estado –sanitarias, criminales, mercantiles – que el usuario las puede hacer suyas y esgrimir las frente a un mal servicio por parte del trabajador. Cuando corresponda, las reglas pueden provenir de las organizaciones gremiales o políticas a las que pertenezca el trabajador, o bien a ámbitos desligados del trabajo para actores no clientes como reglamentos de tránsito, sanitarios, de moral pública, etc.

Es decir, podemos encontrarnos en situaciones de imbricación de reglamentaciones complejas formales e informales, no exentas de contradicciones y a las cuales los actores pueden apelar en los casos de violaciones o para ganar ventajas en la prestación de los servicios.

3.) Posiblemente el concepto que en caso del trabajo ampliado sintetice a los dos conceptos anteriores y añada otros elementos importantes sea el de *construcción social de la ocupación*. Otra vez, hay que buscar su origen en el trabajo asalariado clásico. En este, el concepto de empleo entendido como ocupación de asalariados para un patrón, mucho tiempo ha sido pensado abstractamente como resultado del encuentro entre oferta de trabajo (fuerza de trabajo diría C. Marx) y demanda de esta por las empresas, las variables centrales que supuestamente permitirían explicar el empleo serían el salario y el número de puestos disponibles con respecto de los solicitantes de empleo, así como el número de oferentes de trabajo. Sin embargo, en el trabajo asalariado el arribar a la obtención de un puesto de trabajo puede ser detallado con mayor precisión. Por el lado de la oferta de fuerza de trabajo, esta oferta es de hecho un tipo de acción emprendida por los que desean emplearse y, como todas las acciones sociales, se parte de situaciones que el futuro trabajador no escogió y de sus propias concepciones acerca del trabajo. Parte de cierta estructura de la familia, en ingreso, en jerarquías, en lo que se considera trabajos legítimos, de redes familiares, de amistad, de compadrazgo que permiten llegar a las fuentes de empleo. Además, el oferente de mano de obra llega a solicitar empleo con cierta educación, calificación y experiencia laboral, género, etnia, origen urbano o rural y regional, en cierto momento de su ciclo vital. Por otro lado, se sitúa en estructuras macro que pueden aparecer invisibles para el actor, como coyuntura del crecimiento de la economía, estructuras del mercado de trabajo. Por el lado de la demanda de fuerza de trabajo esta tienen que ver con la microeconomía de la empresa, mercado del producto, ventas, inversiones, exportaciones y la macroeconomía que la impacta (inflación, tasa de cambio, déficits en cuenta corriente). Pero también con la configuración sociotécnica de su proceso productivo (tecnología, organización, relaciones laborales, perfil de la mano de obra, cultura gerencial y laboral) y las estrategias de la gerencia de manejo de personal, de relaciones laborales. Cuando sea el caso pueden influir las políticas sindicales de contratación de personal, por ejemplo el dar preferencia a los familiares de los ya empleados. Sin olvidar las restricciones del mercado de trabajo, las leyes laborales o de seguridad social, las políticas sindicales y los contratos colectivos de trabajo.

En el encuentro entre oferta y demanda de trabajo están involucrados sujetos que tienen intereses uno de ser empleado y el otro de conseguir al empleado adecuado, pero estos no actúan con entera libertad, está limitados o impulsados por estructuras micro, meso y macro como las mencionadas, pero ubicados en estas los actores conciben la relación de trabajo de acuerdo con su experiencia, intereses y carga cultural y el encuentro puede coincidir por los dos lados o a frustrarse.

En los trabajos no clásicos tipo I la diferencia más substantiva es la presencia del cliente en el lugar de trabajo y que el producto o el servicio se genera en el momento del consumo (hay un producto material que se consume en el restaurante o bien es un proceso de venta del mismo). Es decir, la construcción de la ocupación depende también directamente del consumidor, que no contrata al trabajador, pero el ser contratado por la empresa depende de que aquel se esté consumiendo en el acto mismo del trabajo. Es decir, la repercusión del mercado del producto sobre el empleo es directamente repercusión sobre el mercado del trabajo. Esta preferencia del consumidor no se basa solo en precio y en calidad del producto sino también en la atención personal, de tal forma que en la producción social de la ocupación no es posible separar la demanda de trabajo de la demanda del producto o al menos no se dan en dos fases separadas, además la presión por parte del cliente de proporcionar un producto-servicio de calidad y afectividad permanece durante toda la actividad laboral y no forma parte solamente del momento de la contratación del trabajador. Es decir, la construcción social de la actividad es permanente y puede verse coartada por las malas relaciones del trabajador con el cliente, además de la propia gerencia. Por el lado del cliente su demanda de servicio implica costo y calidad del mismo, pero dentro de la calidad está la calidez de la relación con el trabajador y la organización. En unos casos el resultado puede ser de compra venta –compra en un supermercado, servicio tradicional de un banco en sucursal-; en otros, el producto se consume en el lugar de trabajo –hospitales, hoteles, restaurantes- y la calidez forma parte integrante del servicio. Este factor puede alterar la demanda del producto y con esto afectar el empleo.

La construcción social de la ocupación se complica en espacios abiertos sean los trabajadores asalariados o no. La demanda del producto influye directamente en la construcción de este tipo de ocupaciones, en lo inmediato depende de los clientes, pero en lo invisible de la estructura del mercado, la inflación, el tipo de producto, el nivel de ingreso de la población. Es decir, un condicionante de estas ocupaciones es el mercado del producto para sintetizar. En lo micro el trato al cliente puede influir en las ventas y con esto en la ocupación, pero muchos otros agentes pueden ayudar u obstaculizar la constitución de esa ocupación. Primero, las posibles organizaciones gremiales no necesariamente sindicales que pueden permitir o impedir ocuparse, además de los miembros de su comunidad de ocupados o de otras competidoras. En segundo lugar la influencia de los actores no directamente laborales o de su venta, como pueden ser agentes públicos intransigentes o condescendientes, los transeúntes y habitantes de la zona –ciudadanos que protestan por la suciedad, por invadir vías públicas, por delincuencia, por el ruido. No habría que olvidar a los proveedores de insumos o de productos para la venta que pueden presionar sobre lo que se ofrece a los clientes y sobre los precios. Sobre estas construcciones de la ocupación pueden influir las reglamentaciones urbanas o rurales,

sanitarias, laborales para cuando se emplean asalariados, fiscales, los acuerdos corporativos entre organizaciones gremiales y gobiernos. No hay que dejar de lado un concepto ampliado de configuración sociotécnica del proceso de producción que implique al cliente como una dimensión importante.

En los trabajos no clásicos tipo II, en los que se trabaja en espacios cerrados propiedad del trabajador, además de lo señalado para el trabajo asalariado y para clientes, proveedores y mercado del producto, habría que añadir la influencia material y subjetiva de la familia por las posibles contradicciones entre espacio y tiempo de trabajo con respecto del necesario para la reproducción familiar.

En todos los casos, la perspectiva de sujetos implica que las estructuras presionan pero no determinan, que estas presiones pasan por la subjetividad de dichos actores y que la construcción de controles, regulaciones o la propia ocupación implica generar decisiones en interacción con otros sujetos situados también en estructuras y poseedores de capacidad de dar significados.

Finalmente, el problema de la estandarización, rutinización no es exclusiva de las producciones materiales, puede haber estandarización en el trabajo simbólico, por ejemplo la repetición de un show sin interacción con el público, aunque el trabajo de creación-invencción puede estar menos sujeto a la estandarización, el del artista, el del científico (De la Garza, et.al, 2009)

II. El Trabajo cognitivo de diseño de software

El trabajo de diseño de software es un caso de producción eminentemente simbólica objetivada que puede realizarse con la intervención directa del cliente en la producción o no. Además, los símbolos que más importan en esta producción son de carácter cognitivo, sin afirmar que ética, sentimientos y estética no intervengan. Por esto le hemos llamado trabajo cognitivo objetivado. Analizaremos el proceso de trabajo del diseño de software a la medida, que implica la intervención del cliente en diversos momentos del proceso de producción y los fenómenos de resistencia y negociación en el proceso de trabajo.

Podemos sintetizar las etapas del proceso de trabajo en el desarrollo del Software a la medida en las siguientes: a). El cliente contacta a la empresa desarrolladora de software a la medida. Se lleva a cabo una serie de entrevistas entre el cliente y la empresa para determinar la lista de requerimientos del sistema informático y estipular las fechas de entrega de los módulos que integran al mismo; b) Una vez convenidos los requerimientos del sistema, se continúa con el diseño, configuración y elección de la forma del desarrollo del Software, donde participan tanto los líderes de proyecto como los programadores (dependiendo del tipo de organización del trabajo, tamaño de la empresa, experiencia, habilidades y destrezas de los programadores, etc.); c) El Líder de proyecto consensua y negocia con los

programadores (Junior y Senior) los requerimientos a desarrollar, así como los tiempos de entrega; d) El programador elige el “mejor camino” para desarrollar las líneas de código que conforman el algoritmo, una vez que el programador concluye las rutinas que integran los requerimientos, éstos se someten a una renegociación no formal, que en la práctica actúa como “un arreglo social no explícito” para corroborar entre programador y líder que no haya errores de configuración o incumplimiento de los requerimientos, una vez resuelto lo anterior, se presentan las rutinas al tester de operabilidad, quien hace pruebas de calidad, con la finalidad de que se exhiban problemas internos de configuración; d) Implementación en campo y prueba con el cliente/usuario final (Torrissi, 1998). Es el periodo de tiempo en el cual convergen dos momentos:

Es decir, el desarrollo de un sistema informático no se circunscribe sólo al proceso de trabajo que comprende la transformación de los requerimientos del software a desarrollar que le da origen, sino que tiene que ver con la implementación del software y la aceptación o rechazo del mismo por el cliente.

En este sentido, podemos establecer algunas características del *trabajo cognitivo* del software (Chudnovsky, López y Melitsko, 2001). En el desarrollo de software a la medida existe una relación triádica entre clientes, directivos y trabajadores; el producto simbólico se objetiva fuera del productor y el cliente, está compuesto por vocablos de los lenguajes de programación utilizados y el conjunto de algoritmos que conforman los distintos requerimientos que estructuran los módulos del software; existencia de una presión cognitiva en todo el proceso de trabajo por solucionar de la mejor manera acompañada, muchas veces, de la resistencia de los programadores, la negociación, el boicot, el consenso, abiertas o soterradas.

El cliente participa mediante una lista de requerimientos –a veces directamente, en otras indirectamente- que debe cumplir el producto solicitado; el gerente o líder presiona cognitivamente a los trabajadores a que sean creativos y resuelvan de la mejor manera los requerimientos que solicita el cliente, pero los programadores auto-reconocen que son las habilidades y destrezas, la colaboración y el juego, mediados por conflictos y resistencias, lo básico para solucionar un problema planteado. Es decir no es la observancia de metodologías verticales y burocráticas lo que permite crear algoritmos que conforman el software; por el contrario, el proceso de trabajo cognitivo está inmerso en una incertidumbre constante, problemas de configuración, rutinas de código no claras que pueden o no cumplir con los requerimientos. Por otro lado en este tipo de software a la medida, el cliente es un sujeto activo al establecer los requisitos del programa y durante el propio diseño. El conocimiento que significa un programa de software puede almacenarse virtualmente, copiarse, optimizarse su uso, y la reproducción de un programa puede implicar mínimos costos.

El proceso de trabajo del desarrollo de un programa informático podríamos definirlo como “*uno en el cual la materia prima son diferentes tipos de símbolos y el resultado es un conjunto de símbolos que sirven para resolver cierto tipo de problemas*”. Proceso que desde el punto de vista cognitivo estaría conformado por cuatro grandes actividades cognitivas: conceptualización, formalización, procesamiento de datos, e implementación:

- La *conceptualización* es el diseño de concepto-grafías de los distintos requerimientos que solicita el cliente/usuario, requerimientos que debe contener el programa a desarrollar. Para ello se programan entrevistas para acordar con el cliente los requerimientos. Este conjunto de necesidades ocasionalmente -dependiendo de la formalidad de la empresa, tamaño o trayectorias administrativas- son firmadas mediante un contrato o cláusulas de requerimientos.
- La *formalización* de dichas necesidades del usuario, son estudiadas, aprehendidas y modificadas por el analista, el arquitecto o el programador mas diestro en un diseño modularizado (depende de la complejidad de los requerimientos). El análisis orgánico de los requerimientos deviene de conceptualizar el problema y resulta en una formalización del mismo.
- El *procesamiento de datos* trata de seguir una serie de consideraciones lógicas entre los requerimientos trazados en el diseño y la formalización del problema. El diseño se subdivide en módulos que poseen coherencia lógica, y éstos, a su vez, se fragmentan en aplicaciones y éstas se dividen en cadenas o bucles; posteriormente las cadenas se transforman en instrucciones al programador, quien convierte dichas instrucciones en un conjunto de algoritmos (símbolos) que construye a partir del lenguaje de programación que haya utilizado;
- *Implementación*, es la entrega del programa informático e instalación en el sistema del cliente. Las empresas algunas establecen acuerdos de pos-venta, como asesorías en la capacitación de los usuarios finales, mantenimiento durante un periodo de tiempo (Yoguel y Boscherini, 2001).

La definición de categorías cognitivas como fragmentación de tareas no significa que haya una estructura rígida, por el contrario éstas están traslapadas, la fluidez del conocimiento entre las fases del trabajo de diseño están yuxtapuestas, son flexibles. Por ejemplo, es común que quien desempeña un puesto en un proyecto, se traslada a otro puesto en el siguiente proyecto; quien es analista en otro proyecto puede ser programador anteriormente. La flexibilidad cognitiva entre los puestos de trabajo parece estar relacionada con la experiencia, habilidades, destrezas que se poseen para determinado proyecto.

Estas fragmentaciones del proceso de trabajo cognitivo en realidad se yuxtaponen, se traslapan, se integran en la pantalla del programador. Es decir, las complejidades y dificultades metodológicas para aprehender el conjunto de requerimientos en la conceptualización del problema, no suceden en forma aislada, el diseño requiere del conocimiento no sólo del analista, también del comentario del programador. En el procesamiento de datos hay un constante ir y venir entre el diseñador y el programador, coexisten una serie de consensos formales e informales en esta interacción cotidiana en el desarrollo del programa; también interviene la tradición organizacional de la empresa, si implementa continuos acercamientos con el cliente para valorar los avances o los diseños o se subvalora al cliente o usuario final. Ahora bien, considerando que las fronteras entre las fragmentaciones del trabajo en el desarrollo del software se desdibujan, no es menos cierto que cobran fuerza una serie de independencias individuales al interior de dicho proceso. Corresponde a una toma de decisión individual la calidad reflexiva de dicho algoritmo, en la decisión está implícita la voluntad e intencionalidad del programador o programadores que repercute en la coherencia lógica de las cadenas de algoritmos que conforman el módulo. Las decisiones individuales de documentar el procedimiento, de explicar la lógica cognitiva en aquellos algoritmos complejos o no, es una decisión que contiene cogniciones, emociones, valores e incluso estética.; es decir, el proceso de trabajo está embebido de una constelación de significados. Dichas *Configuraciones subjetivas* embeben un conjunto de arreglos, consensos, conflictos, resistencias, relaciones de poder, ya sean individuales o colectivos que se inscriben, producen y reproducen formal e informalmente en las interacciones cotidianas al interior como al exterior del proceso de trabajo.

En torno de los conflictos y resistencias en el trabajo cognitivo, vale preguntarse si hay intentos gerenciales de implementar medidas de estandarización del proceso de trabajo (Yoguel, et.al, 2004). Lo anterior tiene que ver con lo que se ha llamado “la aflicción del software”, es decir, la proliferación de errores de programación y, sobre todo, las dificultades por parte de las gerencias de establecer tiempos estándar de producción. Algunos plantean que la solución de la aflicción del software se resolverá en la medida que se implementen herramientas y métodos que resulten en una mayor estandarización y control en el proceso de producción del software, es decir que la organización del trabajo esté basada en una “administración científica de los tiempos y pensamientos” que pretende la Ingeniería del Software y las demás ingenierías que giran alrededor, como la Arquitectura del Software, la Teoría Cliente-Ordenador, la Teoría de la Usabilidad, la Teoría de la Ergonomía del Software, etc. Sin embargo, existe un conjunto de límites a las intenciones de estandarizar en la producción del software:

- Participación proactiva del cliente en distintas fases del proceso de trabajo.
- Incertidumbre en el proceso.
- Ejecución de las líneas de código (LDC) acorde con las destrezas, habilidades y, un conjunto de reglas informales entre los integrantes de los equipos de trabajo.
- Capacidad de decisión *in situ*, en torno a que método es el más adecuado en la ejecución de rutinas y subrutinas de las LDC que integran un módulo.
- La capacidad de decisión *in situ* implica la existencia de *arreglos sociales* formales e informales. Por ejemplo, el nivel de experticia de quien propone una solución, las destrezas y habilidades cognitivas de quien no necesariamente es un experto pero posee un talento en el discernimiento de la solución requerida.
- En el proceso de trabajo del software la comunicación del como y el porqué se desarrollaron determinadas rutinas de LDC del sistema, está basado también en una doble limitación: por un lado, el aspecto formal de la documentación del algoritmo desarrollado (conocimiento codificado); sin embargo, esta descripción carece normalmente de una serie de explicaciones detalladas, reflexivas (resistencia a documentar el proceso reflexivo). Así, coexiste un doble juego, por un lado, la capacidad de saber-resolver un problema planteado por el programador y la no explicación del saber-resolver, como es el caso de una documentación insuficiente del algoritmo desarrollado, o bien el algoritmo que resolvió el problema está oculto entre “candados” (boicot).
- Entre los programadores concurren consensos y luchas por el saber-hacer que se van transmitiendo no sólo en el proceso de trabajo cara a cara, sino que rompen las fronteras de la empresa e invaden el mundo de la vida del trabajador, mediando una fluidez cognitiva del saber hacer virtualizada, pantalla a pantalla. Donde quien transmite el conocimiento no necesariamente requiere de un programador experto en tiempo real, bien puede ser conocimiento codificado-virtualizado, como parte de las *comunidades simbólicas de trabajadores cognitivos* presentes en Internet que colaboran y participan indirectamente en el proceso de trabajo. En otras palabras estamos en presencia de nuevas formas de fluidez cognitiva y nuevas formas de “arreglos sociales de participación” en el proceso de trabajo.

Particularidades complementarias en el proceso de trabajo del software (Perez, 2003) (David y Forey, 2002) (Da Costa, 2000) serían la:

- Conformación de un conjunto de *constelación subjetivas* que se construyen formal e informalmente; representan intencionalidades y conflictos que pueden ser individuales o

colectivos. Estas son inherentes al proceso de trabajo: subjetividad creativa, subjetividad s gnica y subjetividad que se objetiva.

- o Alto grado de integraci3n del trabajo social en la pantalla del programador, y flexibilidad cognitiva que limita la racionalizaci3n de tareas al estilo taylorista.
- o *rigidez t cnica* en la reorganizaci3n de los programas inform ticos del software con ciclos cortos de cambio tecnol3gico;
- o *inflexibilidad en la socializaci3n del aprendizaje*, es decir presencia de un ciclo relativamente largo para familiarizarse en el uso y manejo del software, que puede llevar a la resistencia del usuario a la adaptaci3n del nuevo software.
- o *Arreglo social de la participaci3n*: comprendemos aquellos intercambios sociales cara a cara, pantalla a pantalla en tiempo real o virtual, de experiencias, habilidades y destrezas que permiten al programador solucionar o imaginar una posible soluci3n algor tmica.
- o *Timig Cognitivo*: Comprendemos una capacidad cognitiva que no est  en funci3n de experiencias, estudios formales o intercambio de conocimiento. Simplemente la excepcionalidad en la soluci3n de problemas (“no todos los programadores pensamos igual”)

Si bien es cierto, el dise o del software se planifica, donde deben destacarse tiempos de finalizaci3n, fechas de implementaci3n del programa (ejecutable para el usuario final), tambi n debe agregarse una estimaci3n de los costos de desarrollo, regularmente estimado en horas-hombre. Sin embargo, estos tiempos, fechas y costos de programaci3n no son cumplidos frecuentemente. La ingenier a del Software no ha logrado establecer metodolog as o m tricas de calidad que sean eficaces, que contemplen retrasos, errores en dise o, fallas del sistema -tanto en desarrollo como en implementaci3n. Otros investigadores, se alan que el conjunto de fallas y errores -considerado como “aflicci3n del software”- es un conjunto de incertidumbres caracter sticas del riesgo en el desarrollo de un programa inform tico. El riesgo forma parte del proyecto (Aurora, Fosturi y Gambardela, 2002). Dice Zahran que el desarrollo de software debe comprenderse como una tarea continua de desaf os continuos. La respuesta a esta aflicci3n del software, por parte de acad micos y empresarios para reducir al m nimo el riesgo y la incertidumbre es la Ingenier a del Software -sobre todo la norteamericana- con un conjunto de metodolog as para verificar procesos de calidad, gesti3n de proyectos, an lisis formal de la arquitectura del software, conceptos y principios del dise o, configuraci3n de software, m tricas de prueba, etc. Sin embargo, a m s de dos d cadas de existencia de la Ingenier a del Software, el proceso de trabajo contin a embebido en riesgos de incumplimiento de los tiempos y costos estimados.

Un complemento de la ingeniería del software, que intentan estandarizar el proceso cognitivo de diseño, llevan a gerentes, y directivos y algunos académicos a considerar que es pertinente impulsar una división del trabajo más estricta en la programación, que sea la gerencia quien realice el diseño, señale los requisitos de los módulos, y sólo encomendar a los programadores el aspecto de la codificación de los algoritmos. Se desea que el área administrativa predomine en el proceso de trabajo, ya que este es un trabajo altamente dependiente del trabajador-programador, para disminuir la incertidumbre no sólo en cuanto al procedimiento, sino también en la documentación de lo que se hizo y porque se hizo tal procedimiento algorítmico. Sin embargo, dice un programador:

“...yo no he tenido que darle cuentas a nadie, entonces, me cuesta trabajo el especificar lo que tengo que hacer (documentar el proceso), por que muchas veces al no tener que rendirle cuentas a alguien, lo vas haciendo conforme ves que te van llegando a veces las ideas,... pero si me preguntas ¿que voy hacer mañana?, igual, todavía no lo defino...” (Programador, JO3-MIXE, Julio de 2006).

Parte de la aflicción del software, reside en el hecho de que los programadores no sólo no documentan el proceso, también hay una alta incertidumbre en cuanto a porque hicieron determinado algoritmo. También puede estar esta presente una resistencia por el control del trabajo al interior del proceso de trabajo del programador, por el control del saber hacer del programador con implicaciones en posibilidades mayores o menores de negociación de éste frente a la gerencia.

Por la persistencia del riesgo e incertidumbre en el desarrollo el software a pesar de la existencia de la Ingeniería del Software, se ha integrado una corriente de académicos y empresarios europeos que critican el método general de la Ingeniería del Software, en el sentido de que esta se ha centrado casi exclusivamente en los atributos y particularidades del software, y ha descuidado al usuario final, se ha centrado más en temas relacionados con el sistema de operación interna, con el rendimiento y fiabilidad del sistema; es decir se ha centrado en aquellos factores que pueden ser medidos objetivamente como: numero de errores por cada numero de líneas de programación; tiempos de respuesta y probabilidades de error por cada determinado numero de código, etc. Es decir, se progresó en aspectos “duros”, deterministas, verticales para intentar solucionar la calidad del proceso de trabajo (Casas, 2003): Métodos y herramientas de análisis, diseño, codificación y pruebas; revisiones de técnicas formales; estrategia de pruebas multi-escalada; control de documentación de software y de los cambios realizados; procedimientos de ajuste a los estándares de desarrollo de software; mecanismos de medida y de información.

Conclusiones

Los trabajos no clásicos, diferentes de la evolución de las manufacturas, pueden implicar la incursión del cliente en el proceso de trabajo y con ello ampliarse los conceptos de control y de relación laboral al menos a un actor más que no es el patrón ni el trabajador (Correa, 1990). Pero también el trabajo no clásico es aquel en el que el componente simbólico –la producción de símbolos- predomina sobre las objetivaciones físicas de los productos. Los productos simbólicos pueden ser objetivadas o subjetivadas. Pero en ambos casos el núcleo de la actividad de trabajar se realiza en la conciencia del trabajador. De tal forma, que el análisis clásico del trabajo como operaciones físicas de transformación, aunque conlleven concepciones, no procede en parte en los trabajos de producción simbólica. Otras disciplinas tendrían que acudir al auxilio de la sociología del trabajo para entender los procesos de producción de símbolos: las teorías cognitivas, las de la producción de la cultura, las de los sentimientos, las estéticas.

En particular, la producción de software no puede desglosarse en operaciones físicas, eventualmente estandarizables. Sino en operaciones lógicas de creación de conocimientos, limitadamente estandarizables. Sin embargo, la producción de conocimiento nunca es pura, significados de otros órdenes siempre intervienen y sobre todo, no se da en el asilamiento individualista, sino en redes de relaciones. Estas redes, no se circunscriben al territorio cerrado de la empresa, sino que se extienden al ámbito familiar y sobre todo a través de redes virtuales de especialistas ubicados en cualquier parte del mundo.

Una característica del trabajo cognitivo del software supone una redefinición de la actividad laboral a nivel de interacción cotidiana entre los actores que participan, como el programador, el cliente y el gerente, interacción que no sucede en un contexto fordista, supeditada a ordenes o cumplir tiempos y movimientos usando las maquinas; es decir, habría una ***construcción social de trabajo ampliado o no clásico*** que se sucede al interior de comunidades simbólicas de trabajo (De la Garza, 2006a:16). Estas *comunidades cognitivas del trabajo de Software*, donde las relaciones laborales se configuran en una serie de contingencias no estrictamente predecibles que las diferencian del proceso de la manufactura. No nos referimos sólo a la posible participación del cliente en el proceso productivo, sino también a la yuxtaposición de las fases del proceso de producción y la objetivación puramente simbólica del producto desarrollado fuera del consumidor y del proceso mismo de producción, así como la acumulación del saber hacer (información y conocimiento) entre dichas comunidades cognitivas.

Las comunidades cognitivas de trabajadores de software a la medida implican la *constelación de configuraciones subjetivas* que se suceden en la interacción dentro y fuera del proceso de trabajo, en forma cara a cara y virtual por un conjunto de trabajadores que establecen relaciones sociales amplias, extendidas hacia el interior como al exterior del proceso de trabajo en tiempo real y virtual, sincrónico y diacrónico; no son estructurales, jerárquicas o rígidas y no están limitadas o restringidas al ámbito laboral; es una comunidad con estructuras que le constriñen y con grados de libertad en la toma de decisiones que implica procesos de construcción de significados a nivel individual o colectiva. *Las comunidades cognitivas de software*, estarían entonces constreñidas por una serie de estructuras que forman parte del proceso de trabajo pero se mantiene la libertad relativa del programador en la toma de decisión en la configuración de las líneas de código que dan origen a los algoritmos. Los algoritmos, representan un conjunto de *signos* que tienen su origen en los lenguajes de programación (existen más de 2000 lenguajes). Los algoritmos son el aspecto lógico formal y están embebidos por diferenciados grados de incertidumbre (dependerá de la complejidad del algoritmo desarrollado). La incertidumbre de un algoritmo está en relación directa con condiciones objetivas y subjetivas. Estas son el grado de conocimiento del lenguaje de programación utilizado, experiencia medida en cantidad de programas desarrolladas en el lenguaje que se utiliza, comprensión metódica del problema planteado y “traducción” correcta de la instrucción en un algoritmo, pero también la habilidad y destreza cognitiva en la resolución de problemas con el lenguaje de programación que se pretende utilizar. En otras palabras es el procedimiento reflexivo para escribir el requerimiento en un código lógico y coherente que hemos denominado como texto *sígnico*.

En el proceso de trabajo del software no se aplican los procedimientos clásicos del trabajo que vienen de la manufactura, el software no está restringido a las “leyes” de la manufactura tradicional o por los límites físicos y temporales de la fábrica o de ritmos y estandarizaciones impuestos por la gerencia propios de la producción en serie del fordismo. El proceso de trabajo del software no es de esfuerzo físico, es reflexivo, es simbólico; la eficiencia y la productividad del trabajo no se ajusta a las economías de escala y aunque las Ingenierías del Software proponen medir la productividad y eficiencia con métricas como son la producción por hora hombre o errores por cada ciento de programas desarrollados, o implementar estándares de calidad como CMM; ISO, IEE, etc., éstas no resultan las más adecuadas por caras, burocráticas y rígidas. Las economías de escalas de la manufactura no aplican en el desarrollo de Software porque el primer software concentra el costo total invertido y en el segundo software clonado su costo tiende a ser equivalente al costo del dispositivo donde se genere la copia. Es decir en el trabajo clásico, los costos forman parte importante de la post-

producción, en cambio en el trabajo del software el costo se concentra en el primer producto. Sin embargo, si se presentan errores en el software post-producción, no significa que sean errores generados en el uso, sino que ya estaban ocultos en el sistema, es decir *no hay software con cero errores*, como pretende la fabricación flexible; el software contiene fallas implícitas al *ciclo de desarrollo* (periodo de tiempo del proceso de trabajo que tarda en desarrollar el software) que se forman en el diseño (conceptualización y formalización de los requerimientos) o bien en el procesamiento de los datos (generación de rutinas de algoritmos o bucles de códigos) y se exteriorizan en las pruebas de calidad o en la implementación (con el cliente o el usuario final). Es importante señalar que el software con errores y defectos no se rechaza, no se destruye (en el mejor de los casos, obviamente dependerá de la magnitud de los mismos), se “parcha” y se asume como inevitable que se incrusten nuevos dispositivos de “actualización”. La falta de métricas y metodologías de calidad eficientes que comprueben no sólo los errores y defectos, sino que evalúen el progreso, el cumplimiento de tiempos y proceso implica que constantemente se sobrepasen los tiempos acordados para la entrega del proyecto, es decir *no hay software justo a tiempo*. Aún con *herramientas procedimentales* de cuarta o quinta generación (programación orientada a objetos y aspectos, métricas de calidad tipo CMM, estándares IEEE, etc.) el proceso de trabajo contiene un alto contenido de “*creado a mano*” en el software a la medida. Este conjunto de inconsistencias, errores, fallos, defectos no terminar a tiempo, etc. están contenido en una serie de contingencias denominada “*aflicción del software*”. Paralelamente a este conjunto de aflicciones no hay una planificación sistemática del proceso de trabajo; escasa aplicación de normas de calidad; insuficiencia o limitación en el seguimiento y gestión del proyecto desarrollados; necesidad de personal con conocimientos, habilidades, destrezas, experiencias en nuevos métodos y plataformas tecnológicas. Así como un entorno polarizado en las calificaciones impartidas por el Estado (heterogeneidad en las profesiones) y en las estrategias que implementan las empresas a nivel individual (polarización industrial en el software a la medida).

El proceso de trabajo del software es atravesado transversalmente por una ausencia de recetas únicas, además de no existir un control total de calidad, cero errores, cero defectos e incumplimiento en un justo a tiempo, se suma la imprevisibilidad de no existir “un solo mejor camino” para desarrollar los proyectos de software. Cada proyecto de software es *sui generis*, con contextos determinados, únicos e impredecibles; cada proceso de trabajo para desarrollar un software posee lógicas propias, internas, que varían entre proyectos. Cada software se enfrenta a contenidos diversos, (requerimientos a la medida), necesidades de actualización y mantenimientos específicos, es decir, que los programas se modifican a lo largo del ciclo de vida. Queda como pregunta si a pesar del componente individual

presente en la creación de software, las condiciones laborales comunes y la comunicación a través de la red contribuye a conformar, junto a aspiraciones, resistencias y satisfacciones comunes, una comunidad y una identidad de quienes laboran en este sector.

Bibliografía

Arora, A., A., Fosfuri, y A., Gambardela (2002), Los mercados de tecnologías en la economía del conocimiento, en Foray D., Sociedad del conocimiento, Revista internacional de ciencias sociales, numero 171, Marzo. Pp. 155-174.

Casas, R., (2003), Enfoque para el análisis de redes y flujos de conocimiento, Luna, M., (Coord.) Itinerarios del conocimiento: formas dinámicas y contenido. Un enfoque de redes. Barcelona: Anthropos-Universidad Autónoma Metropolitana Izt., pp. 19-50.

Chudnovsky, D., López, A. y Melitsko, S. (2001) El sector de software y servicios informáticos en la Argentina: Situación actual y perspectivas de desarrollo, CENIT DT 27/07 .

Correa, C. (1999), "The legal protection of software. Implications for latecomer strategies in newly industrializing economies and middle-income economies" Paris: OECD Development Center, Technical Paper N° 26.

Da Costa, A. P. (2000), "Export Growth and Path-Dependence The Locking-in of Innovations in the Software Industry", 4th International Conference on Technology Policy and Innovation, Curitiba, Agosto.

Davis, P., y Foray D., (2002) Una introducción a la economía y a la sociedad del saber, en Foray D., Sociedad del conocimiento, Revista internacional de ciencias sociales, numero 171, Marzo. Pp. 7-28.

De la Garza, E. (2006) "¿Hacia donde va la Teoría Social?" en Tratado Latinoamericano de Sociología Latinoamericana. Madrid: Anthropos

De la Garza E, et.al (2007) "Crítica de la Razón Parapostmoderna", Revista latinoamericana de Estudios del Trabajo, Año 13, No. 19

De la Garza, E. (2007) "Hacia un Concepto Ampliado de Trabajo" en Teorías Sociales y Estudios del Trabajo. Barcelona: Anthropos

De la Garza, Enrique, et.al (2010) "La Querrela de las Identidades" en Trabajo, Identidad y Acción Colectiva(en proceso de edición). México, D.F.: Plaza y Valdés

De la Garza, E., et.al (2009) "Hacia un Concepto Ampliado de Control, Relación Laboral y Construcción Social de la Ocupación", Revista Iztapalapa, No. 62.

Pérez, C., (2003) Revoluciones tecnológicas, cambios de paradigma y de marco socioinstitucionales, en Aboites, J., y Dutrénit, G., (Coords.) Innovación, aprendizaje y creación de capacidades tecnológicas, México, D.F.: UAM Xochimilco. Pp. 13-45.

Torrise, S. (1998), *Industrial Organization and Innovation. An International Study of the Software Industry* N.Y.: Edward Elgar, Cheltenham.

Yoguel, G. y Boscherini, F., (2001), “El desarrollo de las capacidades innovativas de las firmas y el rol del sistema territorial.”, *Revista Desarrollo Económico*.

Yoguel, G.; Borello, J.; Erbes, A.; Robert, V.; Roitter, S. (2004). *Competencias tecnológicas de los trabajadores informáticos argentinos. Más allá de las restricciones de demanda y oferta*. Littec e-papers.